Beyond Adversarial Learning —
[illegible]

[illegible]

[illegible]

Collaborators: Qixue Xiao, Yufei Chen, Deyue Zhang,
and many others from Qihoo 360

360 [illegible]

[illegible]

360 [illegible] [illegible]

Disekt [illegible] *SecDawgs* CTF [illegible]

xCTF [illegible] [illegible]

2016 [illegible] DARPA Cyber Grand Challenge [illegible]

BULLDOG SECURITY SERVICES

CGC CYBER GRAND_CHALLENGE

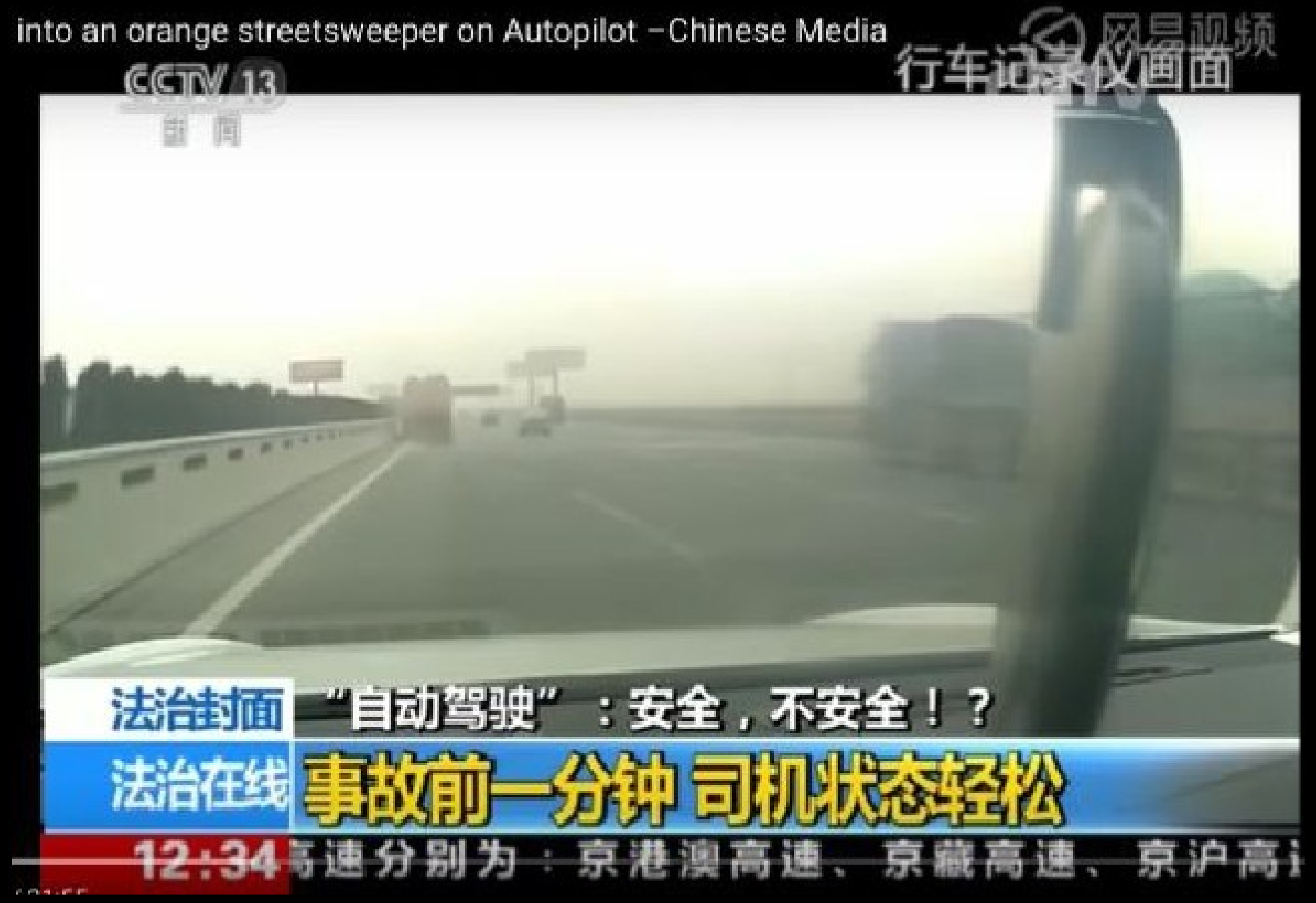
disekt CTF Team

360

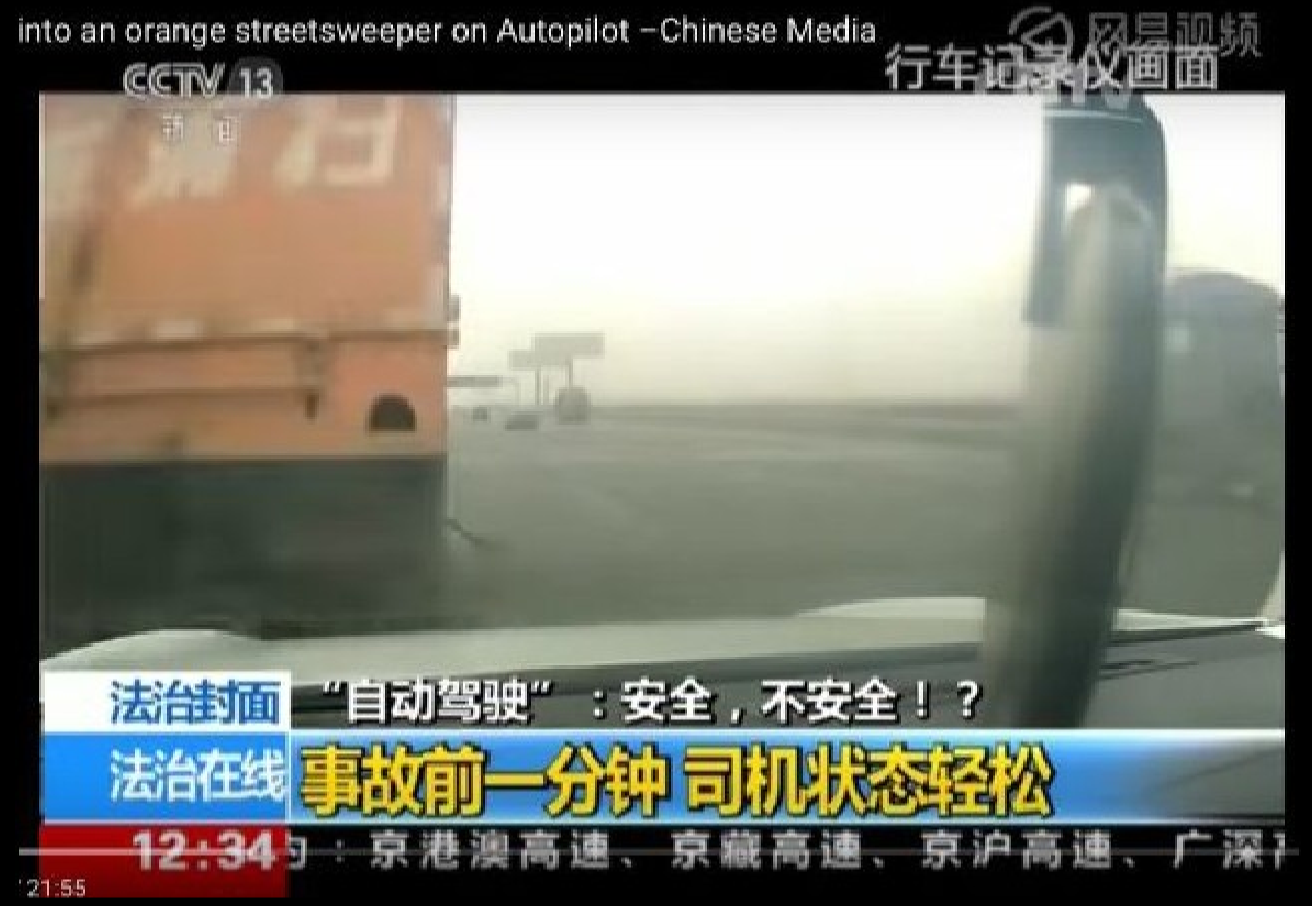
AI [illegible]

[illegible] AI [illegible]?

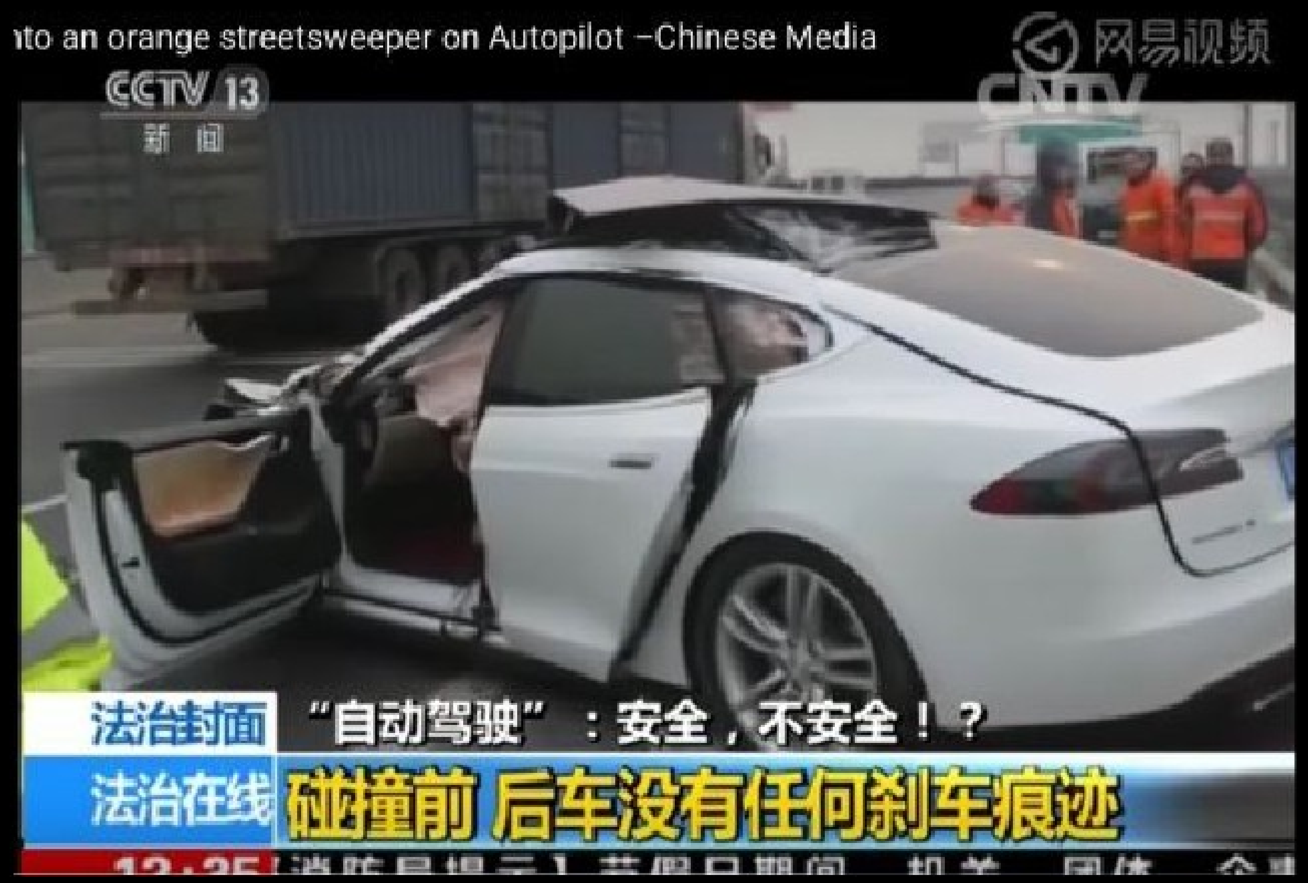
Tesla Crash in China



into an orange streetsweeper on Autopilot –Chinese Media
网易视频
行车记录仪画面
CCTV 13
新闻
法治封面
“自动驾驶”：安全，不安全！？
法治在线
事故前一分钟 司机状态轻松
12:34
高速分别为：京港澳高速、京藏高速、京沪高

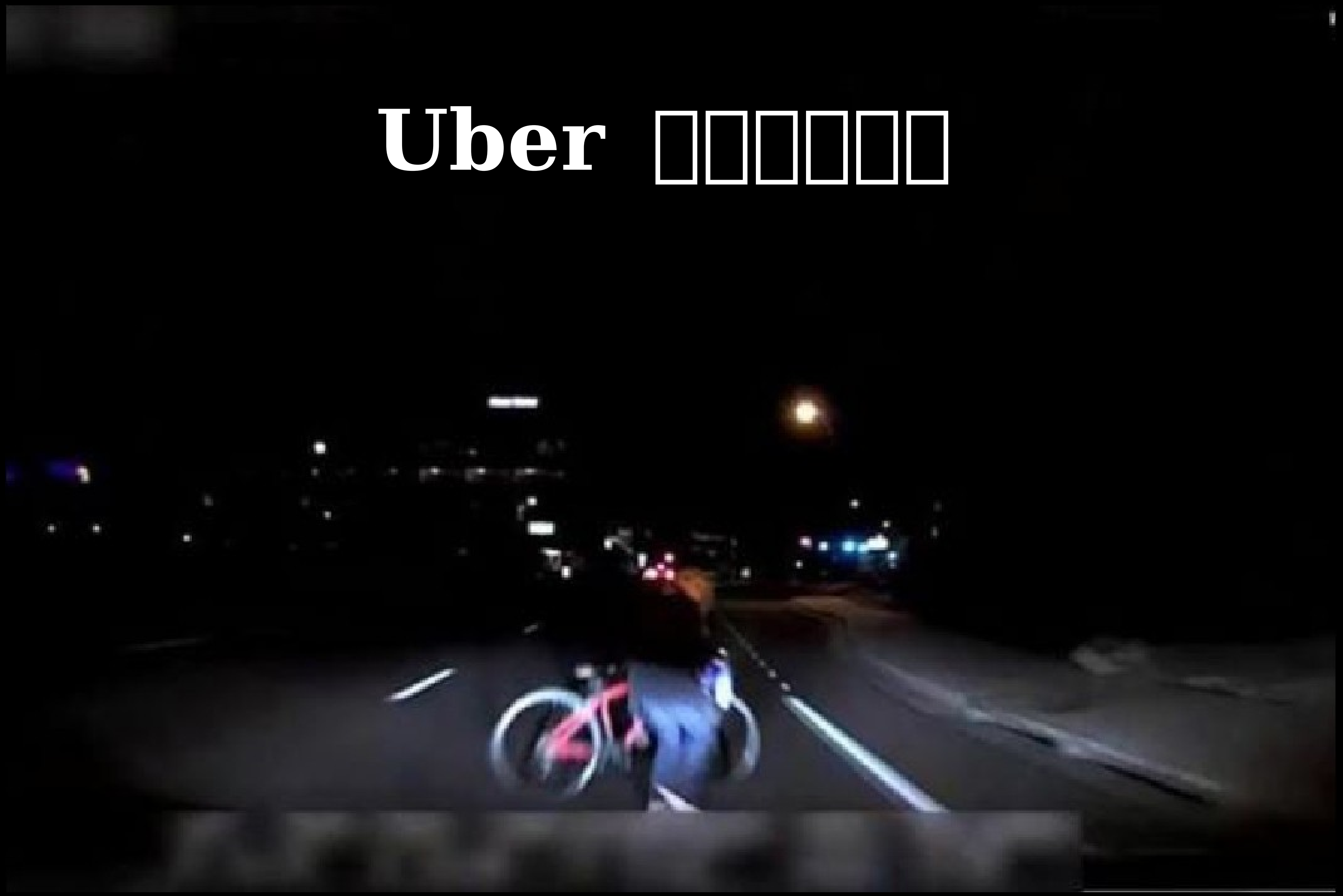


into an orange streetsweeper on Autopilot –Chinese Media
网易视频
行车记录仪画面
CCTV 13
新闻
法治封面
“自动驾驶”：安全，不安全！？
法治在线
事故前一分钟 司机状态轻松
12:34
：京港澳高速、京藏高速、京沪高速、广深
21:55



nto an orange streetsweeper on Autopilot –Chinese Media
网易视频
CCTV 13
新闻
CNTV
法治封面
“自动驾驶”：安全，不安全！？
法治在线
碰撞前 后车没有任何刹车痕迹

Uber [illegible]



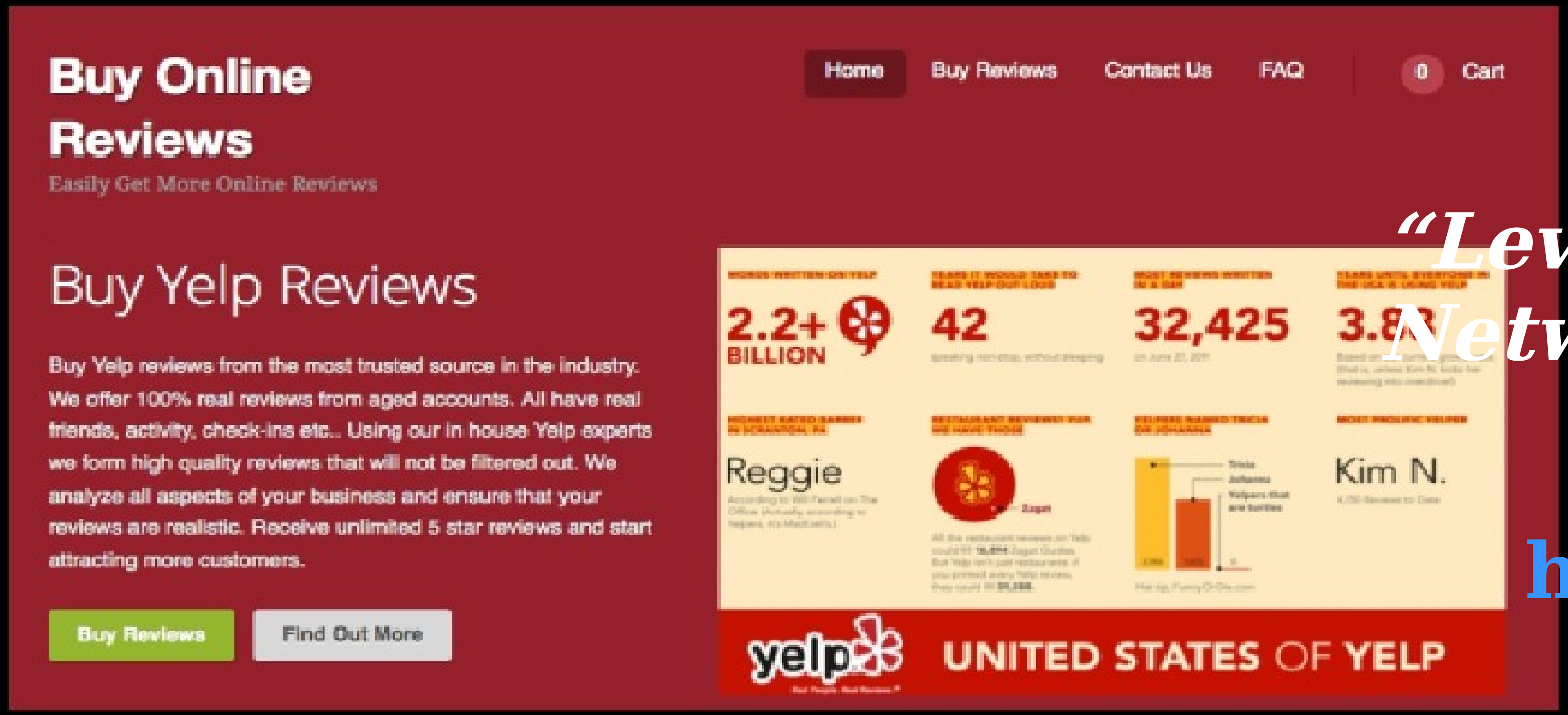
□□□□□□AI□□□□□ (reason #2)

[illegible]“[illegible]”

[illegible]

http://www.caigou.com.cn/news/2017102759.shtml

[illegible]“[illegible]”



Buy Online Reviews

Easily Get More Online Reviews

Buy Yelp Reviews

Buy Yelp reviews from the most trusted source in the industry. We offer 100% real reviews from aged accounts. All have real friends, activity, check-ins etc.. Using our in house Yelp experts we form high quality reviews that will not be filtered out. We analyze all aspects of your business and ensure that your reviews are realistic. Receive unlimited 5 star reviews and start attracting more customers.

[illegible]

“Leverage deep learning language models (Recurrent Neural Networks or RNNs) to automate the generation of fake online reviews for products and services"

https://www.schneier.com/blog/archives/2017/09/new_techniques_.html

[illegible]“[illegible]”

Save or Even Earn Money with Our CAPTCHA Solving Service.

Username: myusername

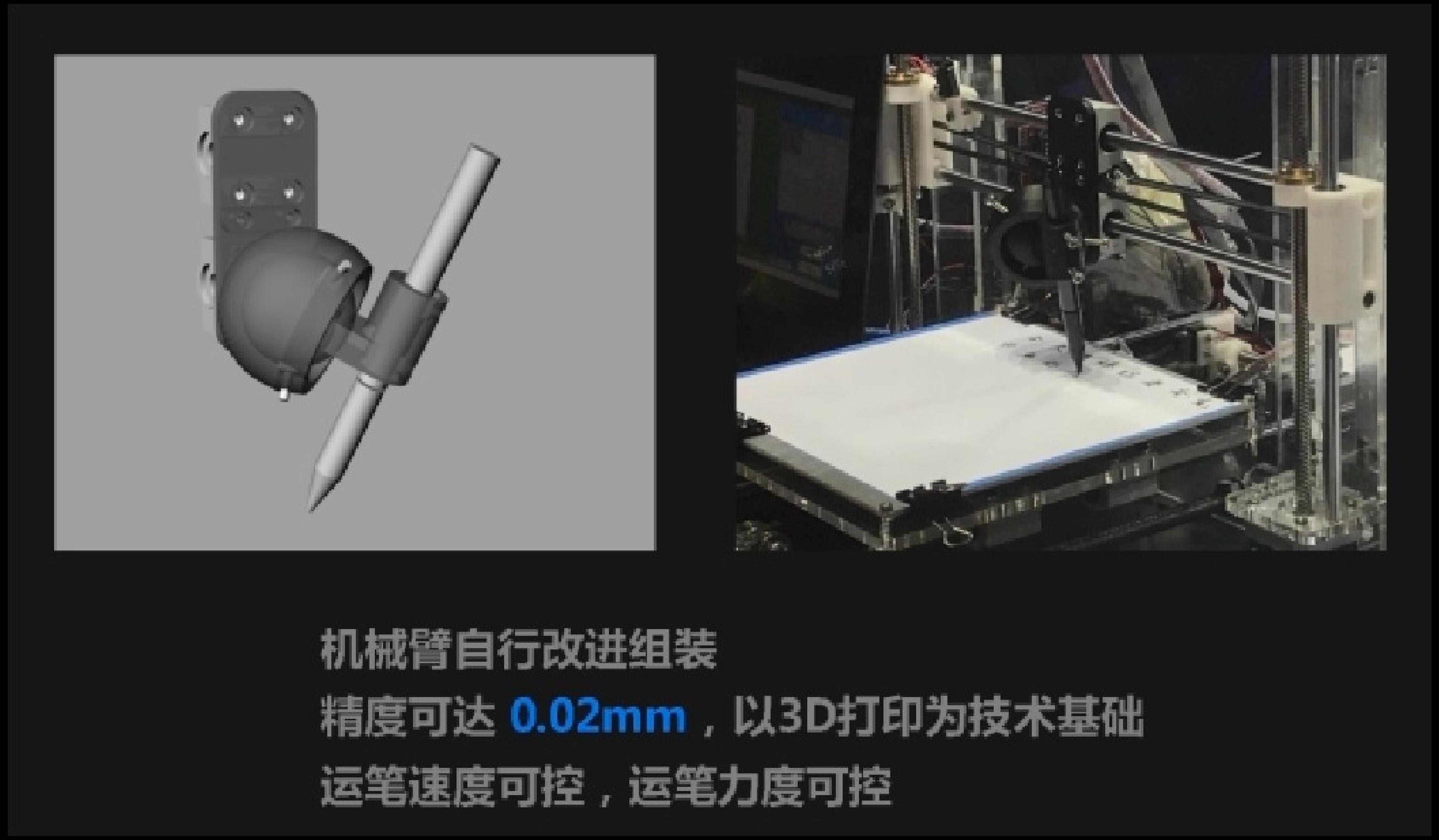
Password: •••••••••••

[illegible]

Defeating Captcha with Learning
https://deepmlblog.wordpress.com/2016/01/03/how-to-break-a-captcha-system/

I’m not a human: Breaking the Google reCAPTCHA
https://www.blackhat.com/docs/asia-16/materials/asia-16-Sivakorn-Im-Not-a-Human

AI [illegible]



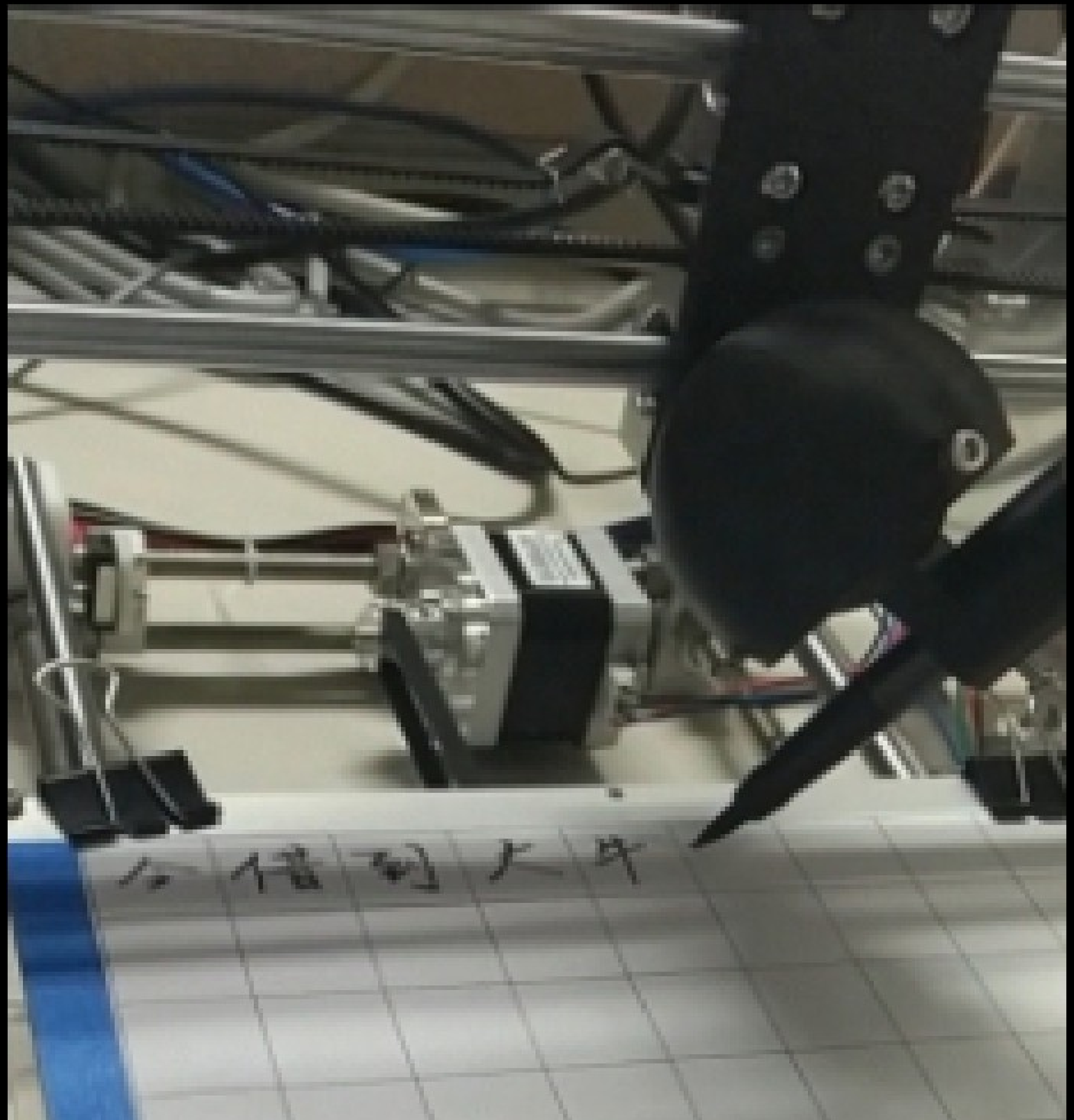
机械臂自行改进组装

精度可达 0.02mm，以3D打印为技术基础

运笔速度可控，运笔力度可控

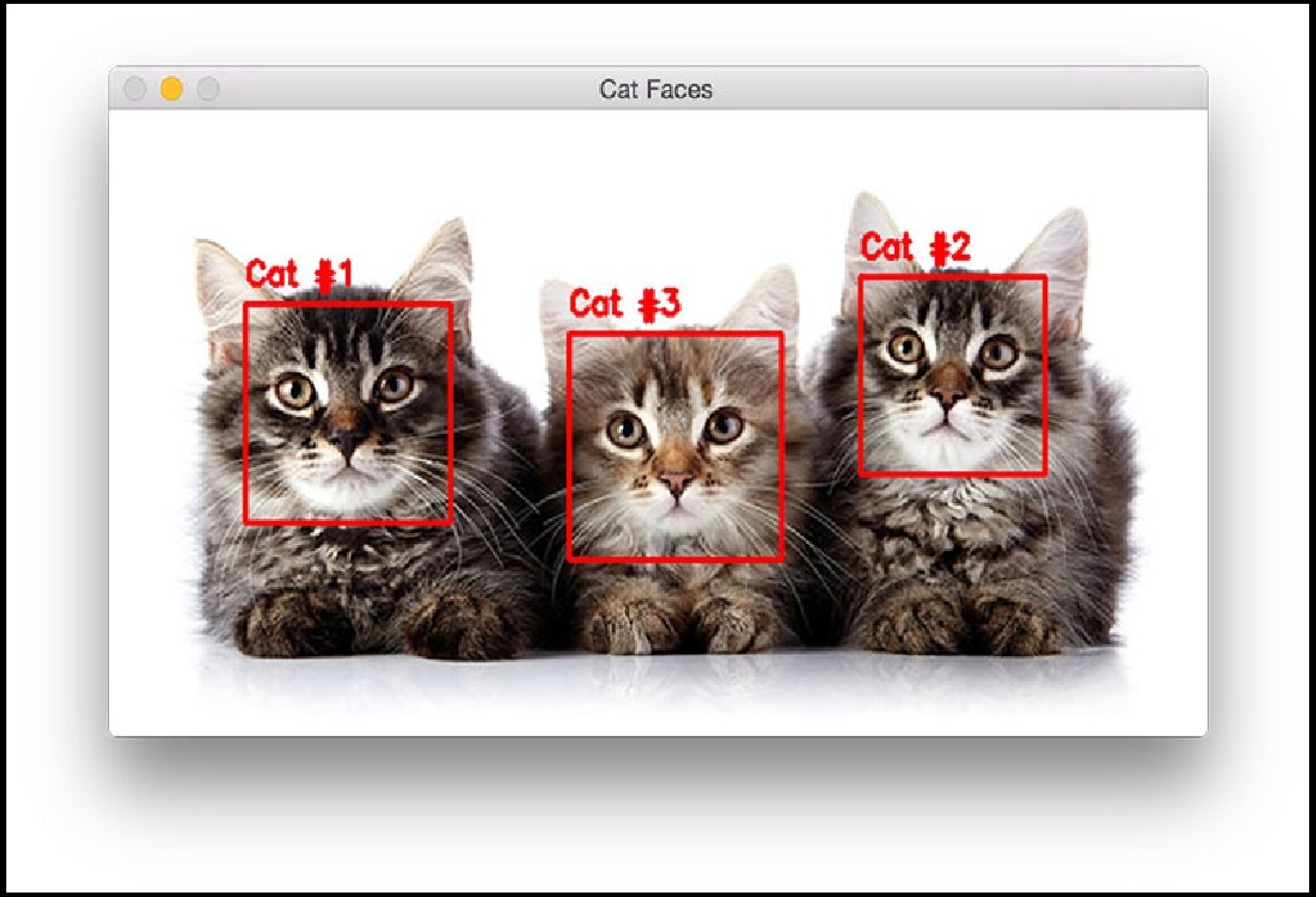
https://2018.qconbeijing.com/presentation/407

Deep Learning-based
Handwriting Generation Machine
— LiChuang@CFCA



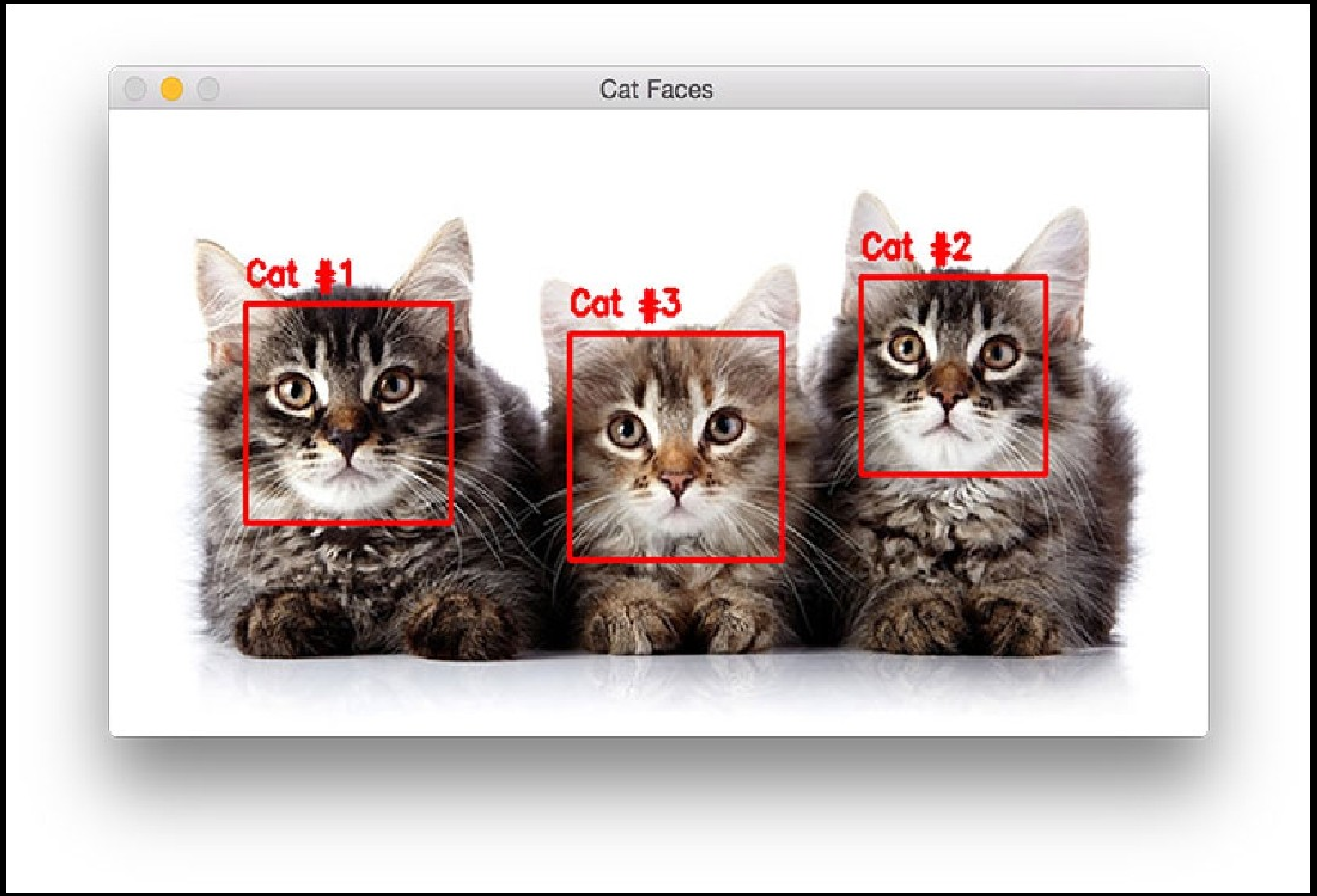
[illegible]

[illegible] / [illegible]



https://www.pyimagesearch.com/2016/06/20/detecting-cats-in-images-with-opencv/

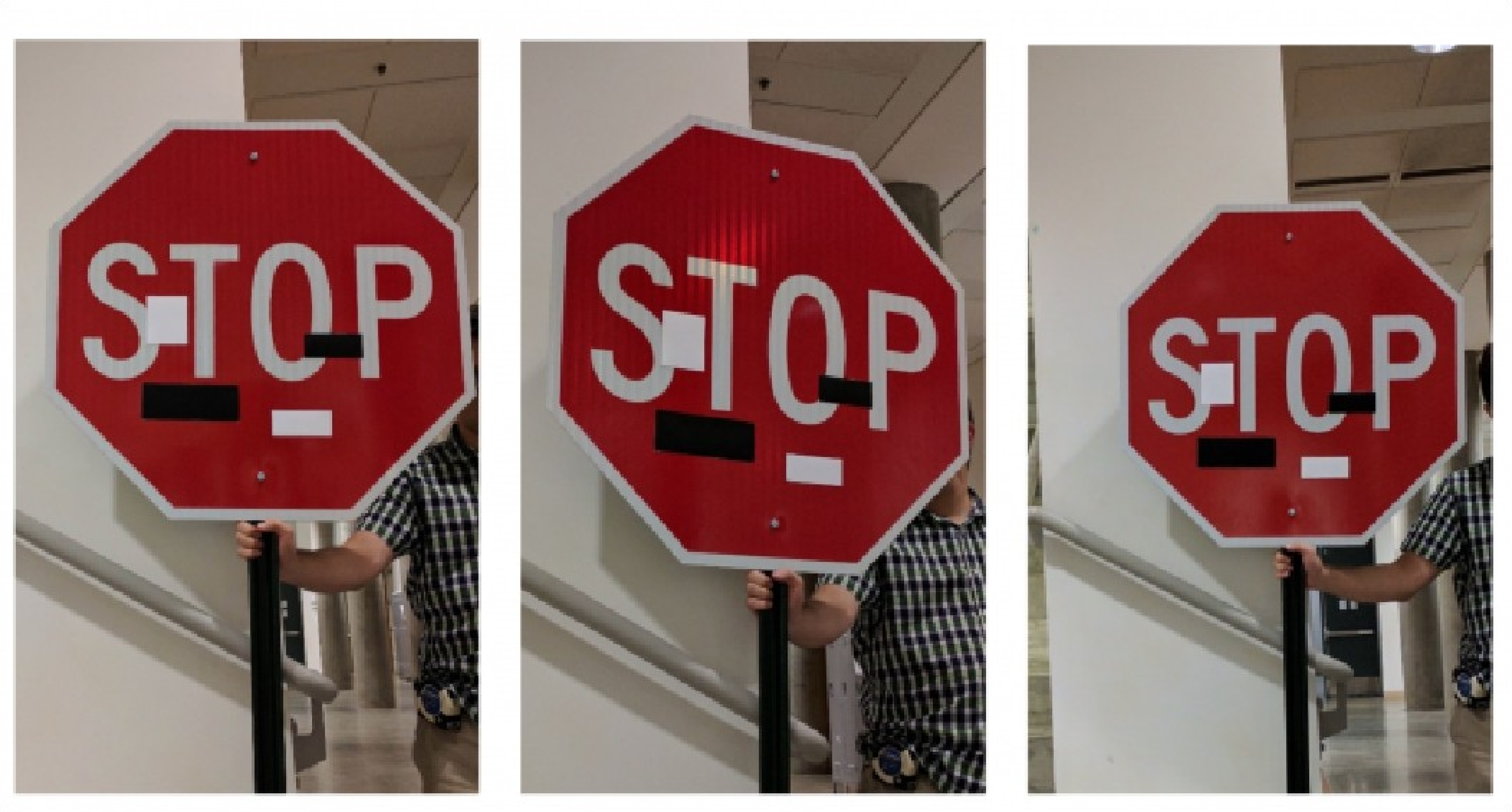
□□□□□□□□□□□□ / □□□□□□□



https://www.pyimagesearch.com/2016/06/20/detecting-cats-in-images-with-opencv/

https://www.wired.com/story/tried-to-beat-face-id-and-failed-so-far/

□□□□□□ □Adversarial Machine Learning□

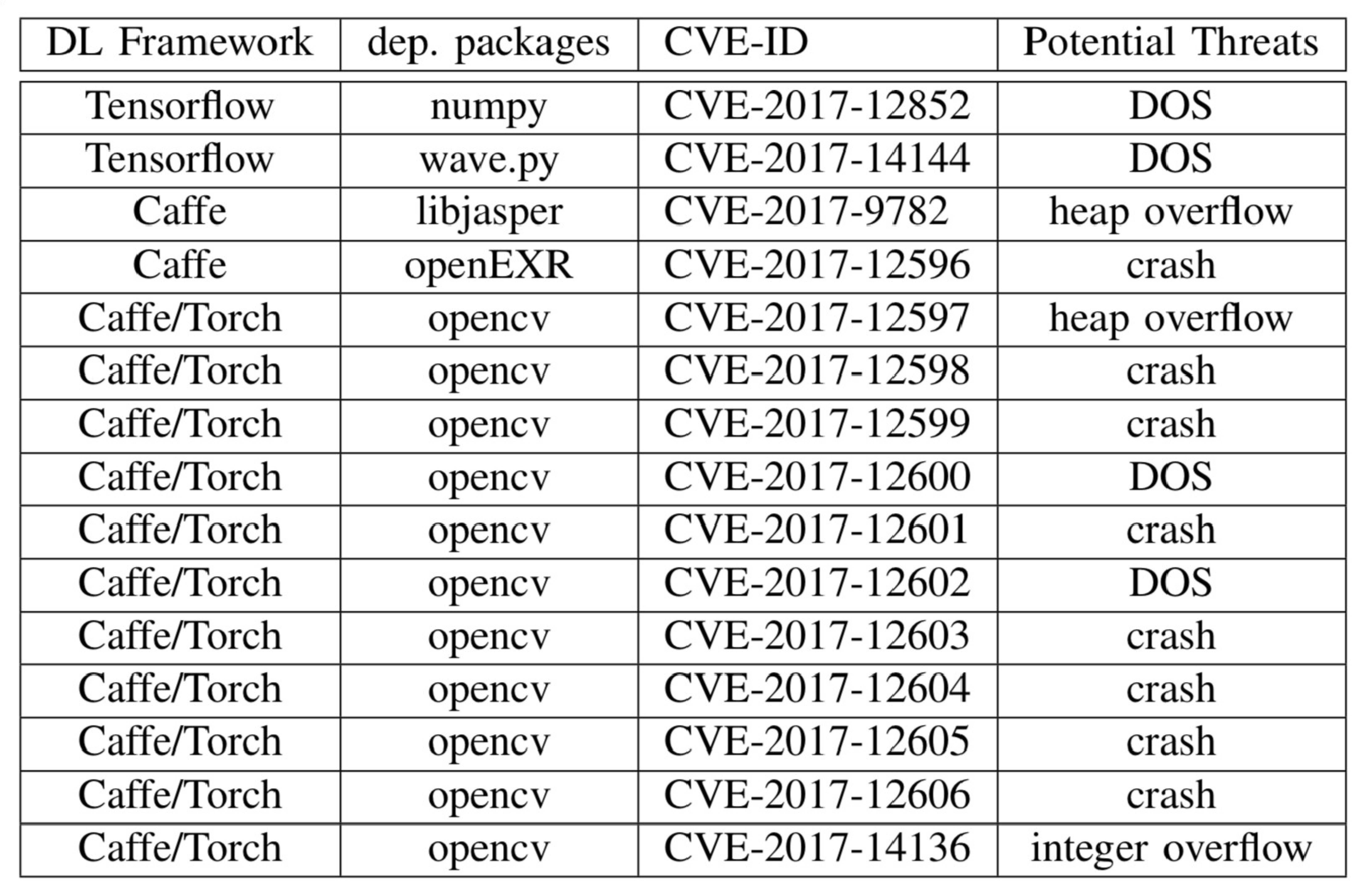


https://medium.com/self-driving-cars/adversarial-traffic-signs-fd16b7171906

Beyond Adversarial Machine Learning

[illegible]

[illegible]



DL Framework	dep. packages	CVE-ID	Potential Threats
Tensorflow	numpy	CVE-2017-12852	DOS
Tensorflow	wave.py	CVE-2017-14144	DOS
Caffe	libjasper	CVE-2017-9782	heap overflow
Caffe	openEXR	CVE-2017-12596	crash
Caffe/Torch	opencv	CVE-2017-12597	heap overflow
Caffe/Torch	opencv	CVE-2017-12598	crash
Caffe/Torch	opencv	CVE-2017-12599	crash
Caffe/Torch	opencv	CVE-2017-12600	DOS
Caffe/Torch	opencv	CVE-2017-12601	crash
Caffe/Torch	opencv	CVE-2017-12602	DOS
Caffe/Torch	opencv	CVE-2017-12603	crash
Caffe/Torch	opencv	CVE-2017-12604	crash
Caffe/Torch	opencv	CVE-2017-12605	crash
Caffe/Torch	opencv	CVE-2017-12606	crash
Caffe/Torch	opencv	CVE-2017-14136	integer overflow

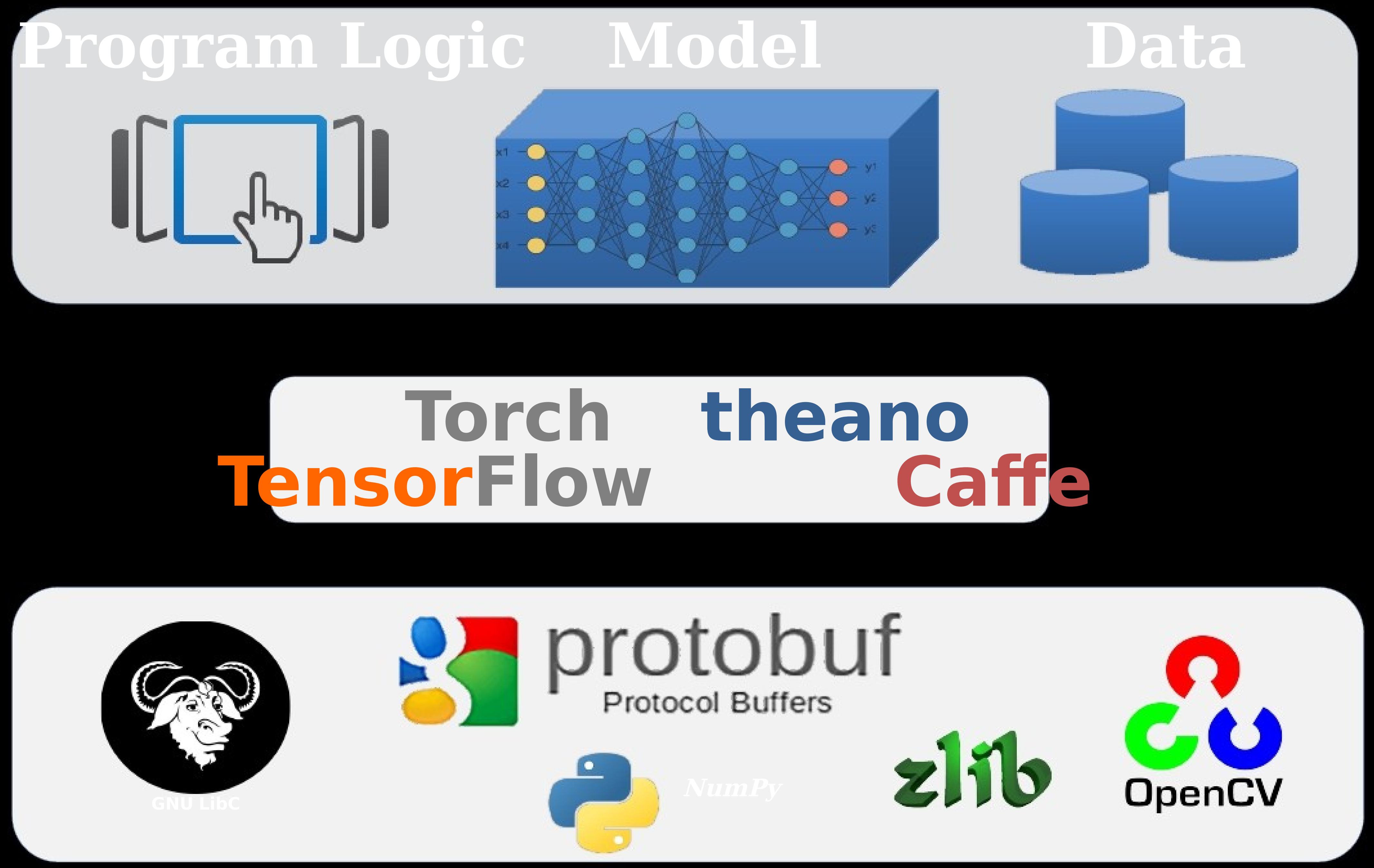
[illegible] 360 [illegible] 2017 [illegible] CVEs

[illegible]

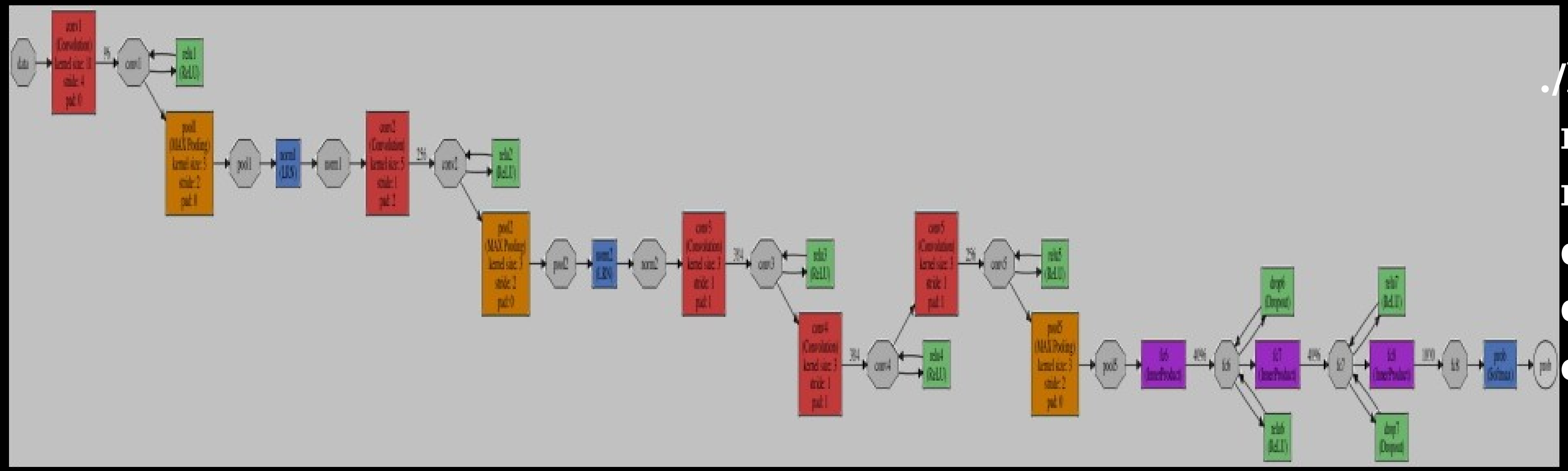
[illegible]

[illegible]

[illegible]


Program Logic
Model
Data
Torch
theano
TensorFlow
Caffe
GNU Libc
protobuf
Protocol Buffers
NumPy
zlib
OpenCV

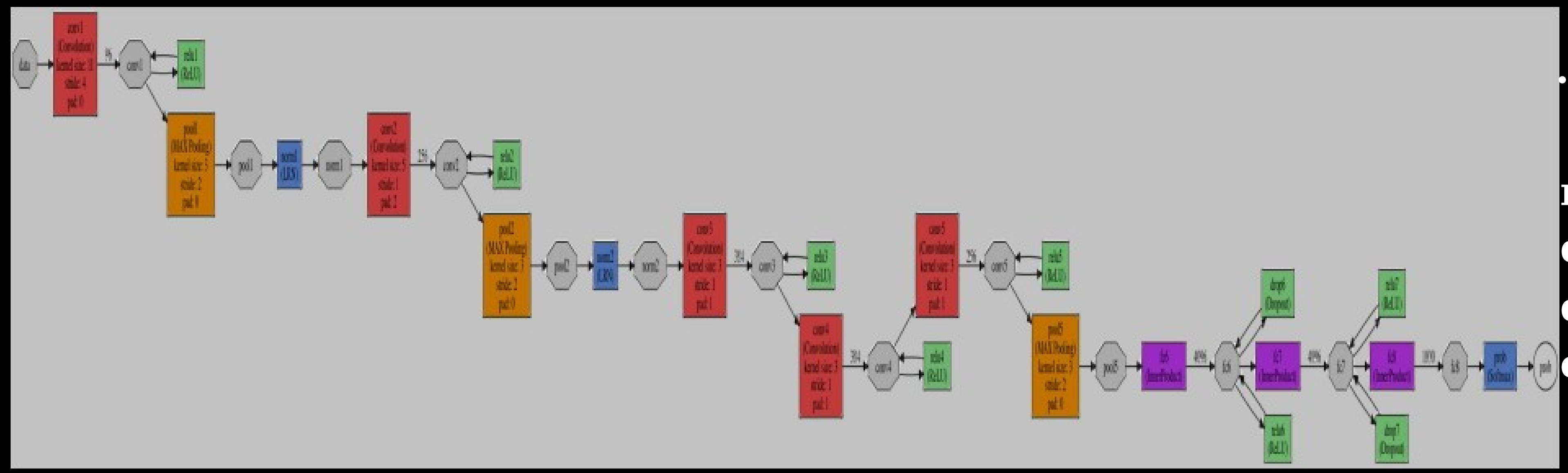
[illegible] (CAFFE [illegible])



```
./build/examples/cpp_classification/classification.bin \
models/bvlc_reference_caffenet/deploy.prototxt \
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel \
data/ilsvrc12/imagenet_mean.binaryproto \
data/ilsvrc12/synset_words.txt \
examples/images/cat.jpg
```

https://github.com/BVLC/caffe/tree/master/examples/cpp_classification

□□□□□□□□□□□ (CAFFE □□)



```
./build/examples/cpp_classification/classification.bin \
  models/bvlc_reference_caffenet/deploy.prototxt \
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel \
data/ilsvrc12/imagenet_mean.binaryproto \
data/ilsvrc12/synset_words.txt \
examples/images/cat.jpg
```

Output:

```
---------- Prediction for examples/images/cat.jpg ----------
0.3134 - "n02123045 tabby, tabby cat"
0.2380 - "n02123159 tiger cat"
  0.1235 - "n02124075 Egyptian cat"
 0.1003 - "n02119022 red fox, Vulpes vulpes"
 0.0715 - "n02127052 lynx, catamount"
```

https://github.com/BVLC/caffe/tree/master/examples/cpp_classification

```
# bullog [Original Image]
./classification.bin models/bvlc_reference_caffenet/deploy.prototxt
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog
---------- Prediction for ./poc_samples/bulldog —————
0.5111 - "n02108915 French bulldog"
```

```
# Threat Example 2 -- Evasion attack
# bulldog_sh [ cause classification to misclassify to an arbitra
# [Here the classification produced a class "Flying Pig" made
./classification.bin models/bvlc_reference_caffenet/deploy.pro
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffe
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_fp
---------- Prediction for ./poc_samples/bulldog_fp ----------
   0.98 - "n03770679 flyingpig”
```

```
# Threat Example 1 -- DoS attack
# bulldog_crash [ cause classification binary to crash]
./classification.bin models/bvlc_reference_caffenet/deploy.prototxt
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_crash
---------- Prediction for ./poc_samples/bulldog_crash ----------
Segmentation fault (core dumped)
```

```
 # Threat Example 3 -- Exploitation Attack
# bulldog_sh [ cause classification to generate a local shell]
./classification.bin models/bvlc_reference_caffenet/deploy.pr
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffe
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_sh
---------- Prediction for ./poc_samples/bulldog_sh ----------
$ uname -a
Linux ctf-box 4.4.0-31-generic #50~14.04.1-Ubuntu SMP
Wed Jul 13 01:07:32 UTC 2016 x86_64 x86_64 x86_64 GNU/Li
     $ exit
```

```
# bullog [Original Image]
./classification.bin models/bvlc_reference_caffenet/deploy.prototxt
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog
---------- Predicti          _samp          log ——————
0.5111 - "n0210          French bulldog"
```

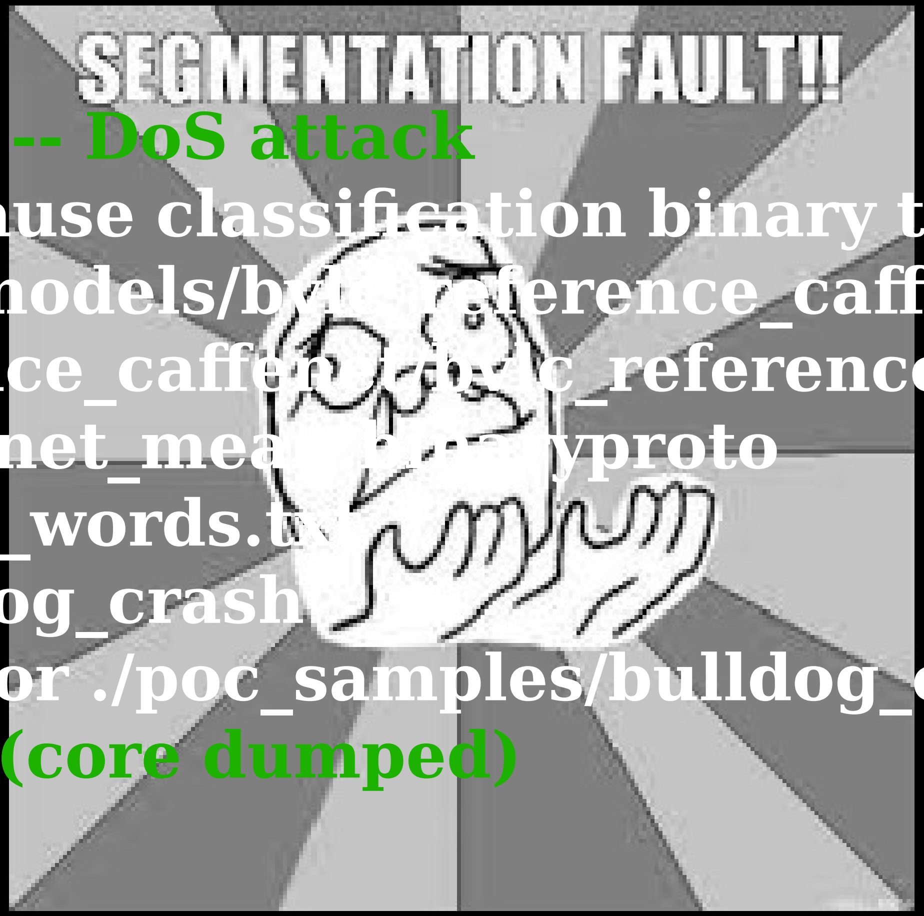
```
# Threat Example 2 -- Evasion attack
# bulldog_sh [ cause classification to misclassify to an arbitra
# [Here the classification produced a class "Flying Pig" made
./classification.bin models/bvlc_reference_caffenet/deploy.pro
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffe
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_fp
---------- Prediction for ./poc_samples/bulldog_fp ----------
   0.98 - "n03770679 flyingpig”
```

```
# Threat Example 1 -- DoS attack
# bulldog_crash [ cause classification binary to crash]
./classification.bin models/bvlc_reference_caffenet/deploy.prototxt
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_crash
---------- Prediction for ./poc_samples/bulldog_crash ----------
Segmentation fault (core dumped)
```

```
 # Threat Example 3 -- Exploitation Attack
# bulldog_sh [ cause classification to generate a local shell]
./classification.bin models/bvlc_reference_caffenet/deploy.pro
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffe
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_sh
---------- Prediction for ./poc_samples/bulldog_sh ----------
$ uname -a
Linux ctf-box 4.4.0-31-generic #50~14.04.1-Ubuntu SMP
Wed Jul 13 01:07:32 UTC 2016 x86_64 x86_64 x86_64 GNU/Li
      $ exit
```

```
# bullog [Original Image]
./classification.bin models/bvlc_reference_caffenet/deploy.prototxt
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog
---------- Predicti            _samp          log ——————
0.5111 - "n0210       French bulldog"
```

```
# Threat Example 2 -- Evasion attack
# bulldog_sh [ cause classification to misclassify to an arbitra
# [Here the classification produced a class "Flying Pig" made
./classification.bin models/bvlc_reference_caffenet/deploy.pro
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffe
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_fp
---------- Prediction for ./poc_samples/bulldog_fp ----------
   0.98 - "n03770679 flyingpig”
```

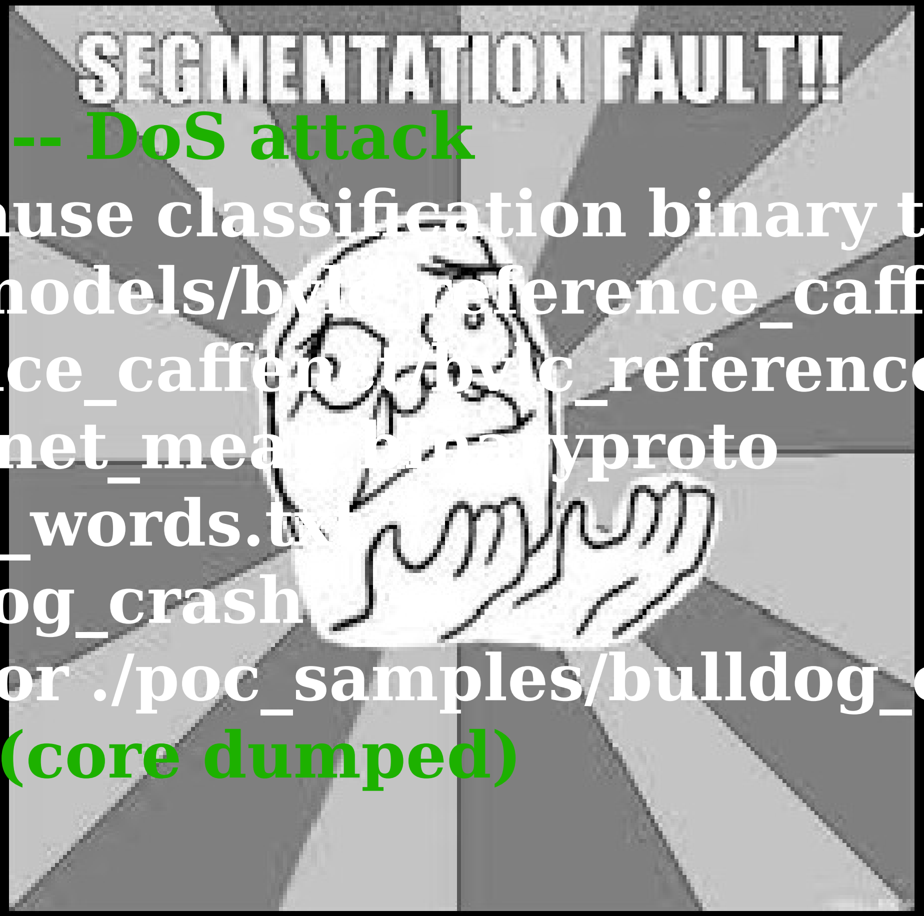

SEGMENTATION FAULT!!

```
# Threat Example 1 -- DoS attack
# bulldog_crash [ cause classification binary to crash]
./classification.bin models/bvlc_reference_caffenet/deploy.prototxt
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_crash
---------- Prediction for ./poc_samples/bulldog_crash ----------
Segmentation fault (core dumped)
```

```
 # Threat Example 3 -- Exploitation Attack
# bulldog_sh [ cause classification to generate a local shell]
./classification.bin models/bvlc_reference_caffenet/deploy.pro
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffe
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_sh
---------- Prediction for ./poc_samples/bulldog_sh ----------
$ uname -a
Linux ctf-box 4.4.0-31-generic #50~14.04.1-Ubuntu SMP
Wed Jul 13 01:07:32 UTC 2016 x86_64 x86_64 x86_64 GNU/Li
    $ exit
```

```
# bullog [Original Image]
./classification.bin models/bvlc_reference_caffenet/deploy.prototxt
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog
---------- Predicti          _samp    /    log ————————
0.5111 - "n0210          French bulldog"
```

```
# Threat Example 2 -- Evasion attack
# bulldog_sh [ cause classificatio      misclassify to   n arbitra
# [Here the classification produc      lass "Flying P g" made
./classification.bin models/bvlc_r        nce_caffenet/deploy.pro
models/bvlc_reference_caffenet/b     eference_caffe et.caffe
data/ilsvrc12/imagenet_mean.bi        oto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_fp
---------- Prediction for ./poc_samples/bulldog_fp ----------
    0.98 - "n03770679 flyingpig”
```

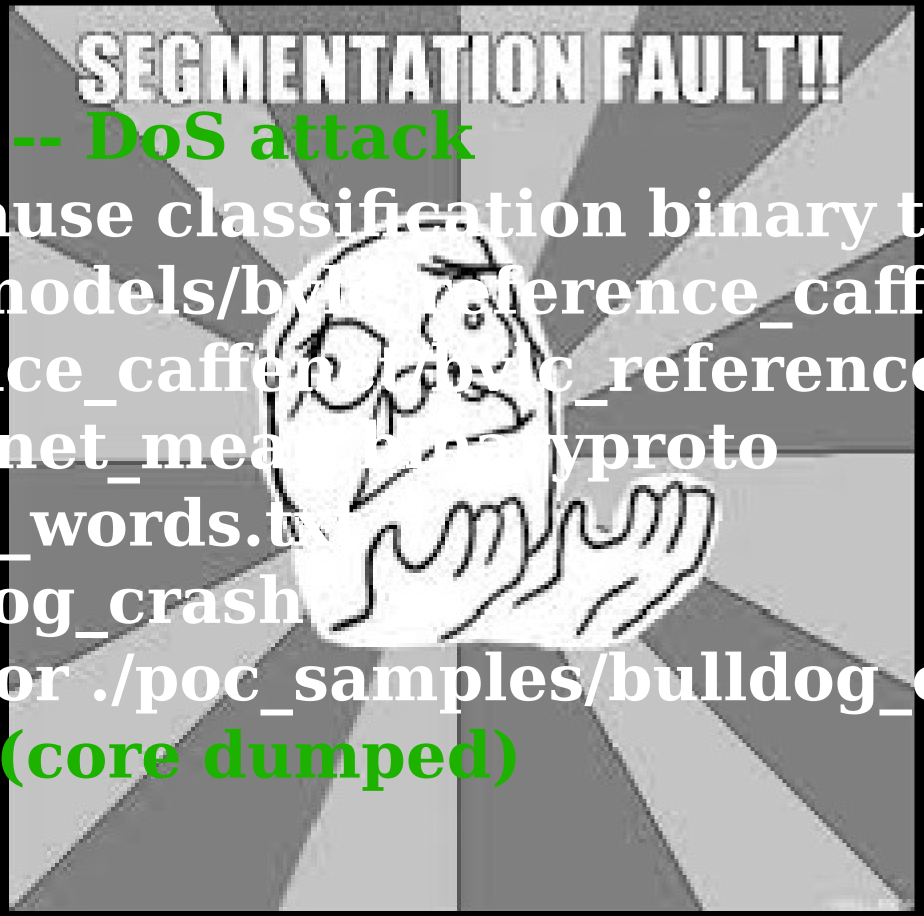

SEGMENTATION FAULT!!

```
# Threat Example 1 -- DoS attack
# bulldog_crash [ cause classification binary to crash]
./classification.bin models/bvlc_reference_caffenet/deploy.prototxt
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_crash
---------- Prediction for ./poc_samples/bulldog_crash ----------
Segmentation fault (core dumped)
```

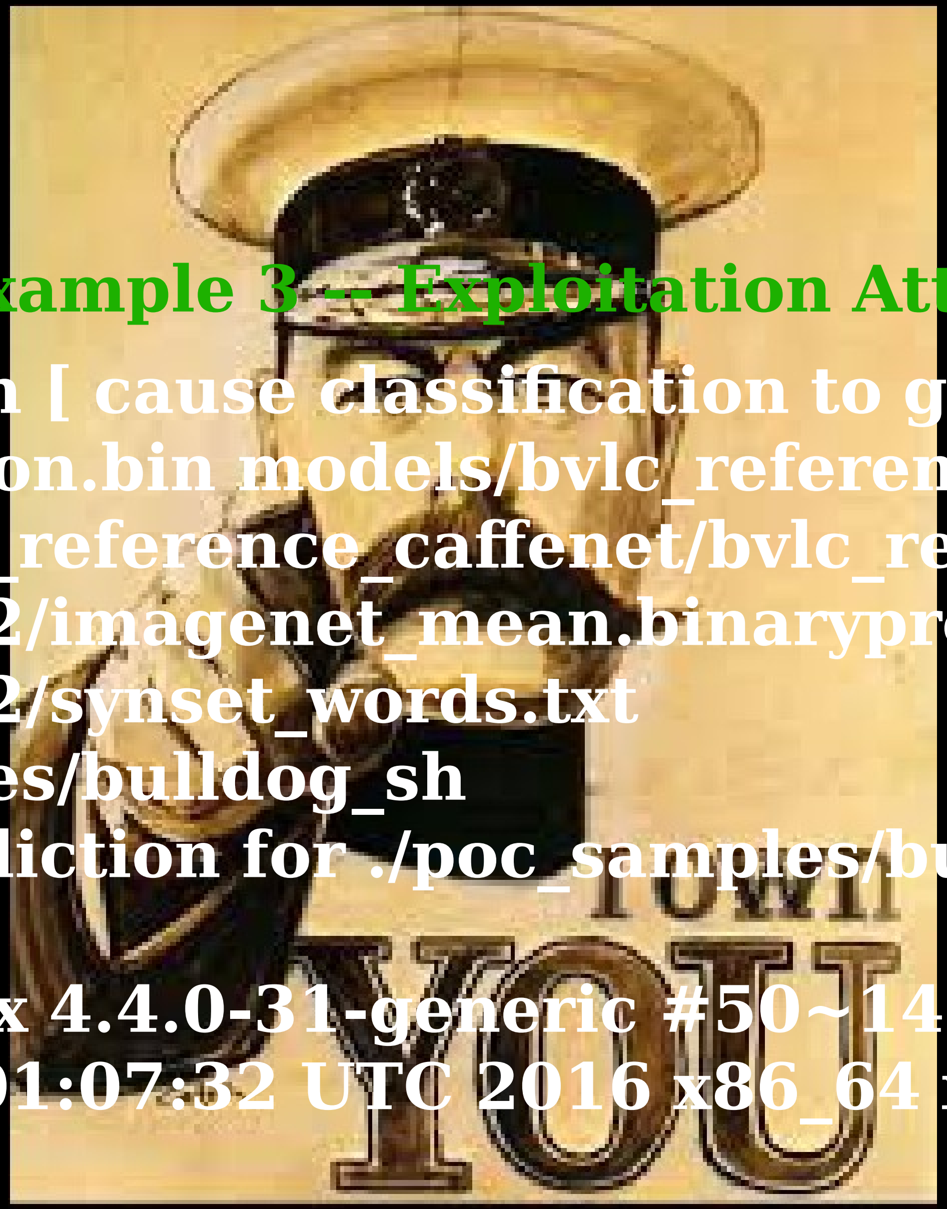
```
 # Threat Example 3 -- Exploitation Attack
# bulldog_sh [ cause classification to generate a local shell]
./classification.bin models/bvlc_reference_caffenet/deploy.pro
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffe
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_sh
---------- Prediction for ./poc_samples/bulldog_sh ----------
$ uname -a
Linux ctf-box 4.4.0-31-generic #50~14.04.1-Ubuntu SMP
Wed Jul 13 01:07:32 UTC 2016 x86_64 x86_64 x86_64 GNU/Li
    $ exit
```

```
# bullog [Original Image]
./classification.bin models/bvlc_reference_caffenet/deploy.prototxt
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog
---------- Predicti[illegible]_samp[illegible]log —————
0.5111 - "n0210[illegible] French bulldog"
```

```
# Threat Example 2 -- Evasion attack
# bulldog_sh [ cause classificatio[illegible] misclassify to[illegible]n arbitra
# [Here the classification produc[illegible]class "Flying P[illegible]g" made
./classification.bin models/bvlc_r[illegible]nce_caffenet/deploy.pro
models/bvlc_reference_caffenet/b[illegible]eference_caffenet.caffe
data/ilsvrc12/imagenet_mean.bi[illegible]oto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_fp
---------- Prediction for ./poc_samples/bulldog_fp ----------
   0.98 - "n03770679 flyingpig"
```


SEGMENTATION FAULT!!

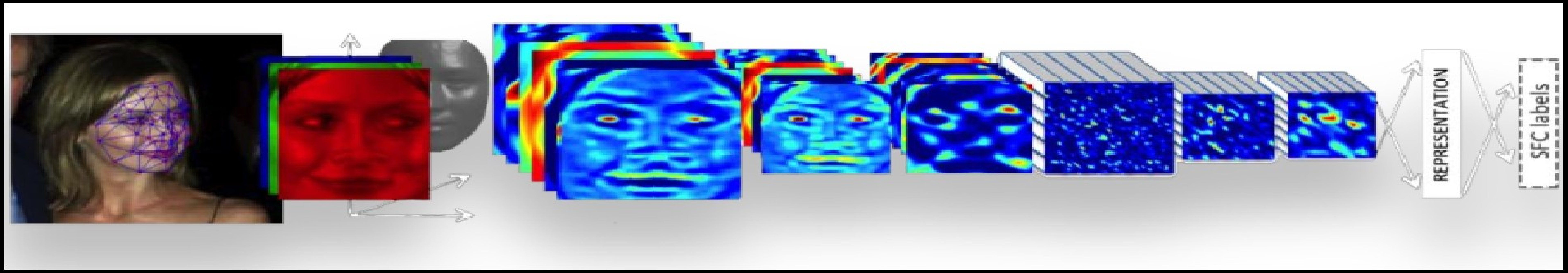
```
# Threat Example 1 -- DoS attack
# bulldog_crash [ cause classification binary to crash]
./classification.bin models/bvlc_reference_caffenet/deploy.prototxt
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffemodel
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_crash
---------- Prediction for ./poc_samples/bulldog_crash ----------
Segmentation fault (core dumped)
```



```
 # Threat Example 3 -- Exploitation Attack
# bulldog_sh [ cause classification to generate a local shell]
./classification.bin models/bvlc_reference_caffenet/deploy.pr
models/bvlc_reference_caffenet/bvlc_reference_caffenet.caffe
data/ilsvrc12/imagenet_mean.binaryproto
data/ilsvrc12/synset_words.txt
./poc_samples/bulldog_sh
---------- Prediction for ./poc_samples/bulldog_sh ----------
$ uname -a
Linux ctf-box 4.4.0-31-generic #50~14.04.1-Ubuntu SMP
Wed Jul 13 01:07:32 UTC 2016 x86_64 x86_64 x86_64 GNU/Li
    $ exit
```

[illegible]

[illegible]

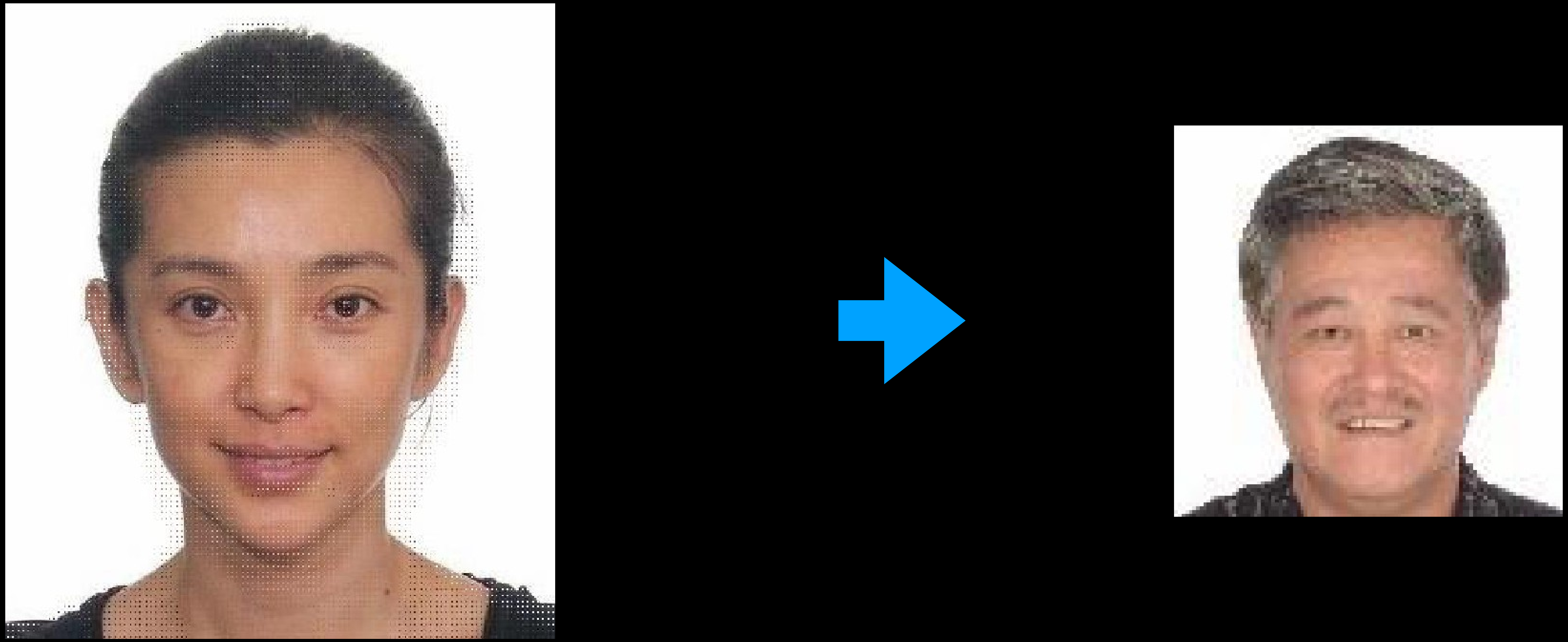


https://www.cs.toronto.edu/~ranzato/publications/taigman_cvpr14.pdf

[illegible]

AI [illegible]

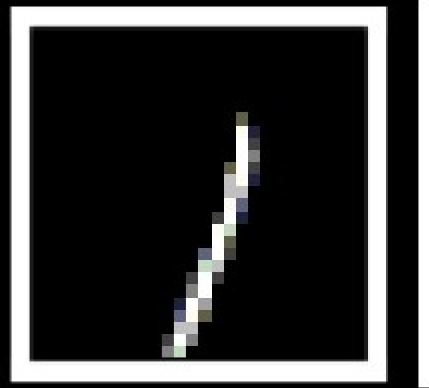
[illegible]

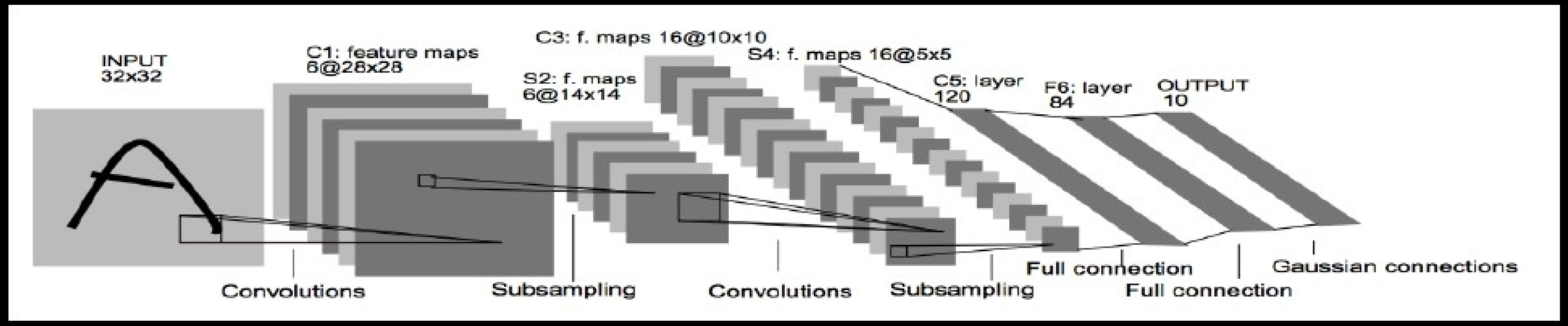


[illegible]

[illegible]

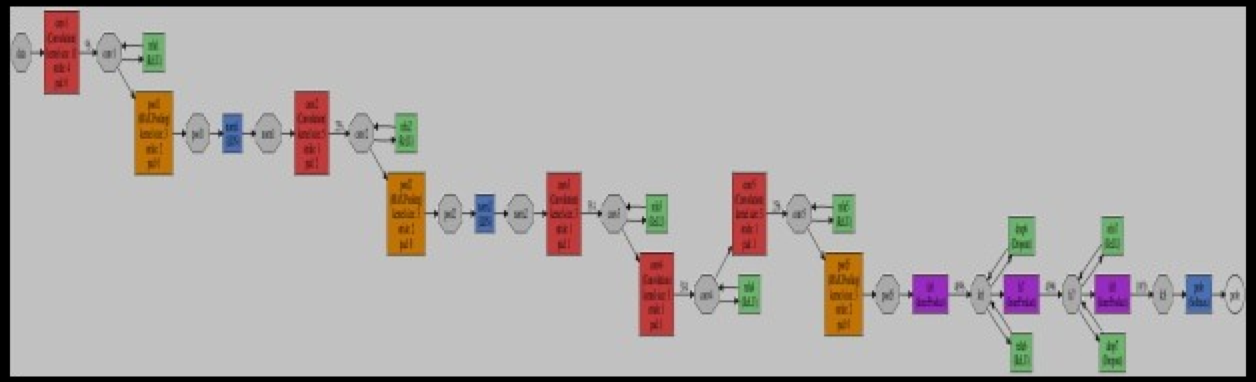
MNIST





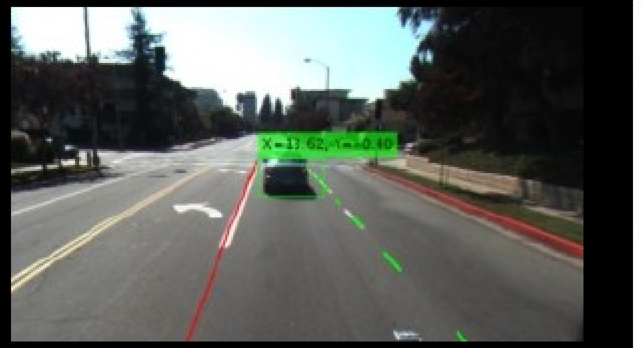
http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf

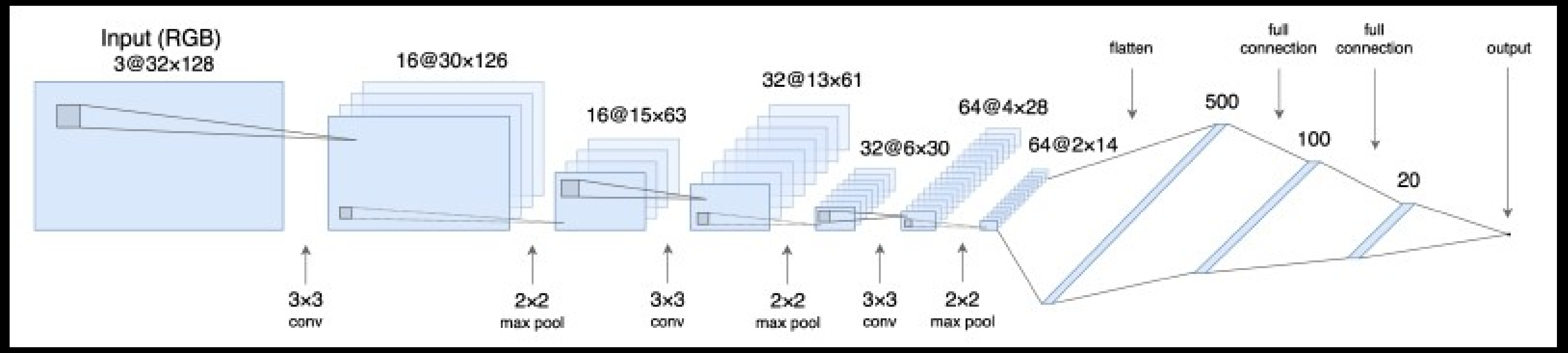
ImageNet



https://github.com/BVLC/caffe/tree/master/examples/cpp_classification

NVIDIA PX DAVE-2





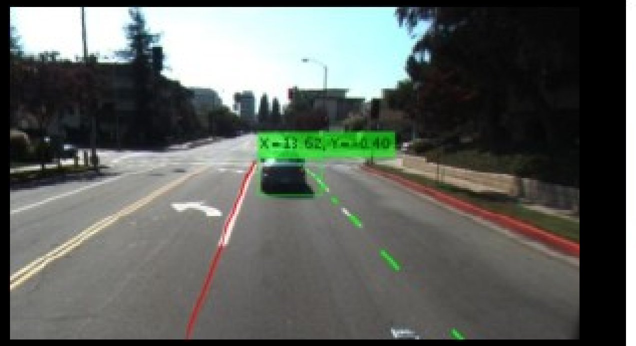
https://images.nvidia.com/content/tegra/automotive/images/2016/solutions/pdf/end-to-end-dl-using-px.pdf

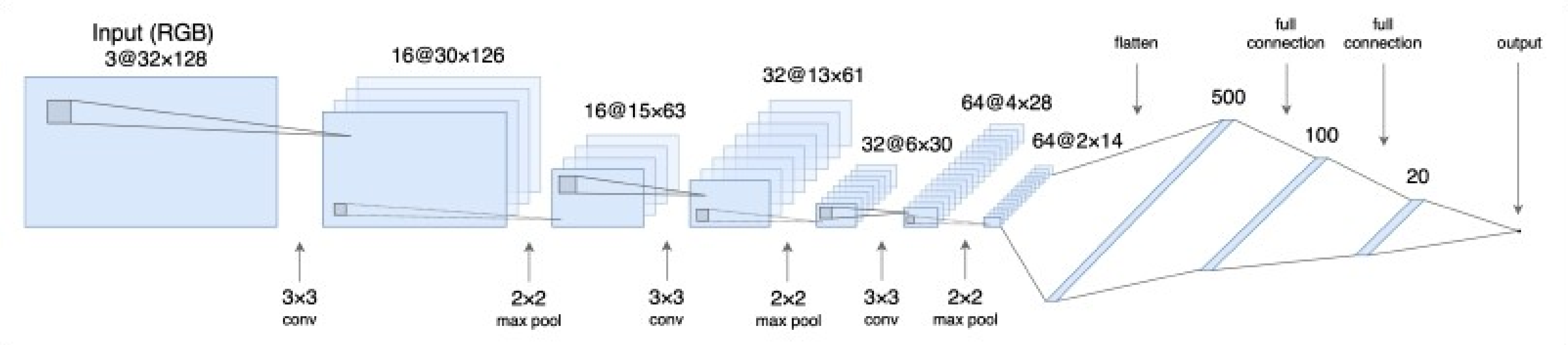
Deep Learning Model Input Requirement (pixel x pixel)

MNIST	28x28
ImageNet	
AlexNet	227x227
GoogleNet,VGG	224x224
ResNet	224x224
NVIDIA	
DAVE-2Self-Driving	200x66

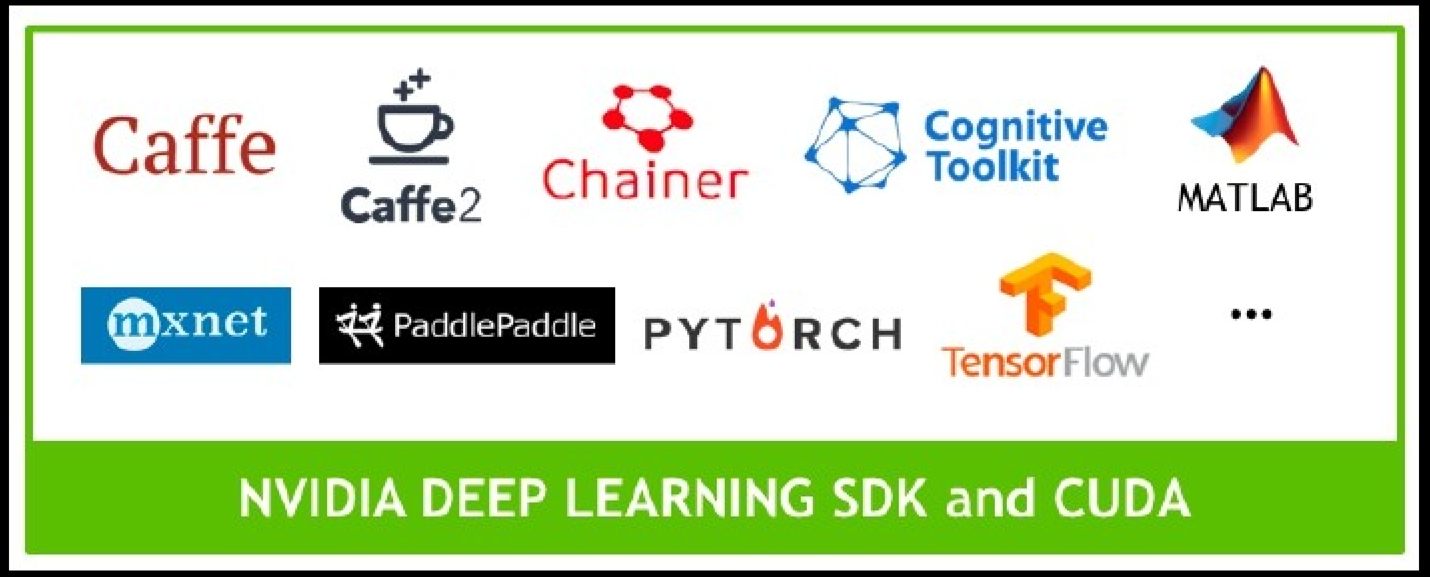
NVIDIA □□□□□□□□□□□□□□□□□□

NVIDIA Sample Self-Driving Models





NVIDIAPX2	
DAVE-2Self-Driving	200x66



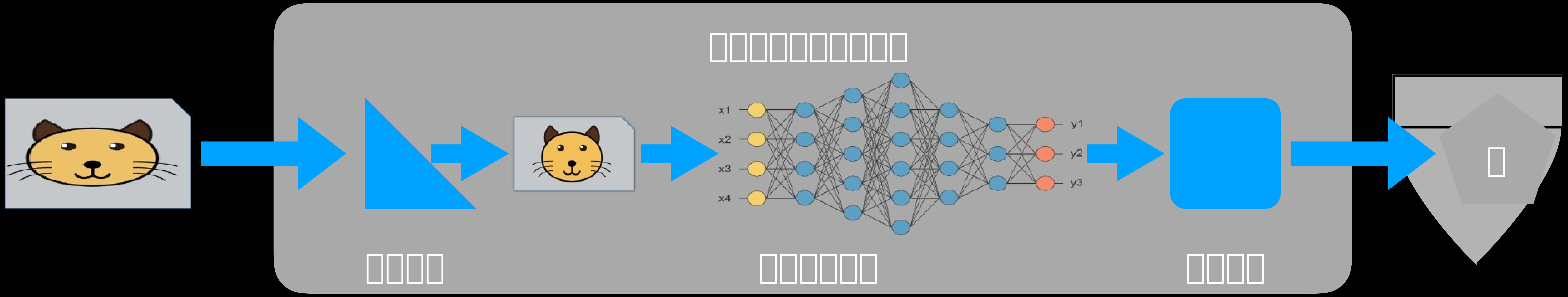
NVIDIA Recommended Ecosystem Camera Vendors

Filr	
A310	320x240
A615	640x480
Leopard	
LI-AR0231	1920x1208
SEKONIX	
SF3326-100	1920x1208

https://images.nvidia.com/content/tegra/automotive/images/2016/solutions/pdf/end-to-end-dl-using-px.pdf

https://developer.nvidia.com/drive/ecosystem

[illegible]


x1
x2
x3
x4
y1
y2
y3

[illegible]

TensorFlow Example

```
def read_tensor_from_image_file(file_name, input_height=299, input_width=299,
                                input_mean=0, input_std=255):
    input_name = "file_reader"
    output_name = "normalized"
    file_reader = tf.read_file(file_name, input_name)
    if file_name.endswith(".png"):
        image_reader = tf.image.decode_png(file_reader, channels = 3,
                                           name='png_reader')
    elif file_name.endswith(".gif"):
        image_reader = tf.squeeze(tf.image.decode_gif(file_reader,
                                                      name='gif_reader'))
    elif file_name.endswith(".bmp"):
        image_reader = tf.image.decode_bmp(file_reader, name='bmp_reader')
    else:
        image_reader = tf.image.decode_jpeg(file_reader, channels = 3,
                                            name='jpeg_reader')
    float_caster = tf.cast(image_reader, tf.float32)
    dims_expander = tf.expand_dims(float_caster, 0);
    resized = tf.image.resize_bilinear(dims_expander, [input_height, input_width])
    normalized = tf.divide(tf.subtract(resized, [input_mean]), [input_std])
    sess = tf.Session()
    result = sess.run(normalized)

    return result
```

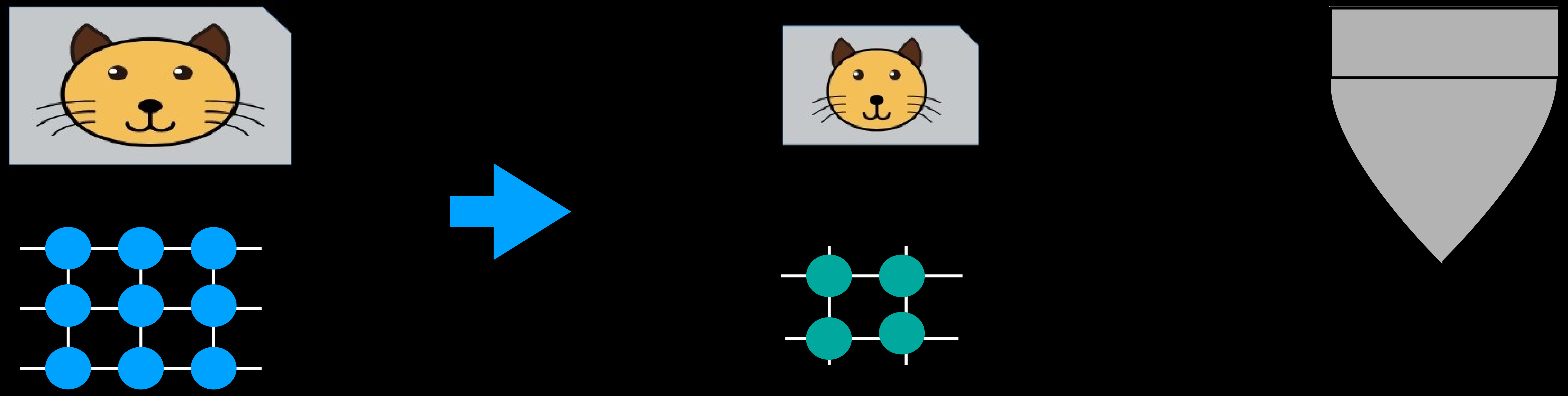
□□□□□□□□□□□□□□□

DeepDetect Example

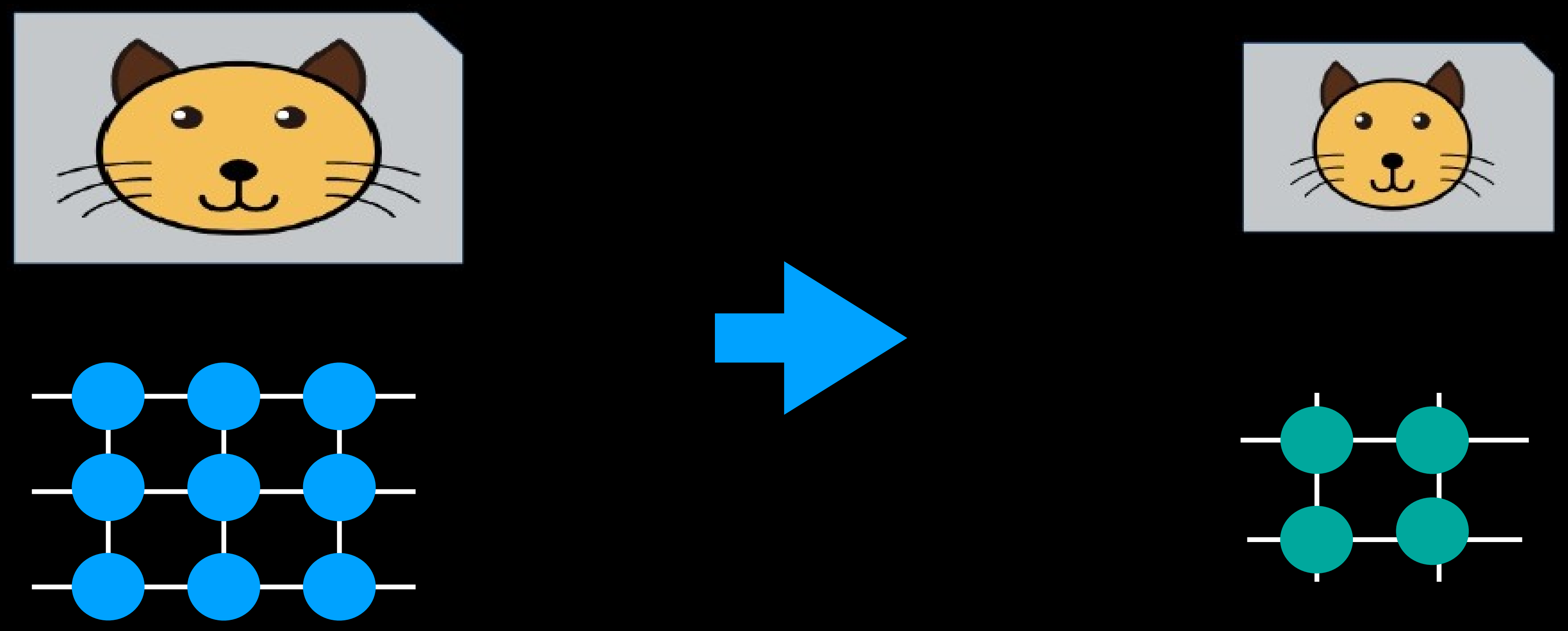
```
int read_file(const std::string &fname)
{
  cv::Mat img = cv::imread(fname,_bw ? CV_LOAD_IMAGE_GRAYSCALE :
                                       CV_LOAD_IMAGE_COLOR);
  if (img.empty())
{
  LOG(ERROR) << "empty image";
  return -1;
}
  _imgs_size.push_back(std::pair<int,int>(img.rows,img.cols));
  cv::Size size(_width,_height);
  cv::Mat rimg;
  cv::resize(img,rimg,size,0,0,CV_INTER_CUBIC);
  _imgs.push_back(rimg);
  return 0;
}
```

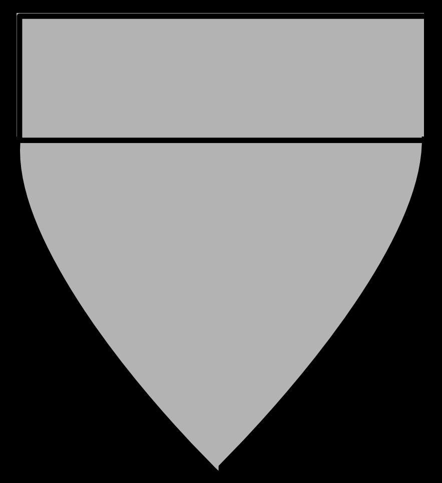
□□□□□□□□□

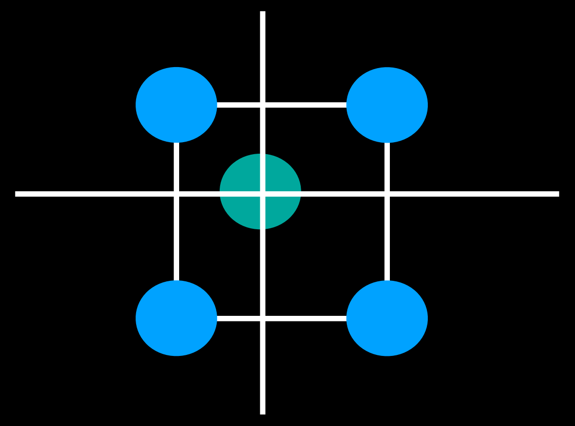
Scaling is supposed to preserve the visual features of an image and thus does not change its semantic meaning.



[illegible]

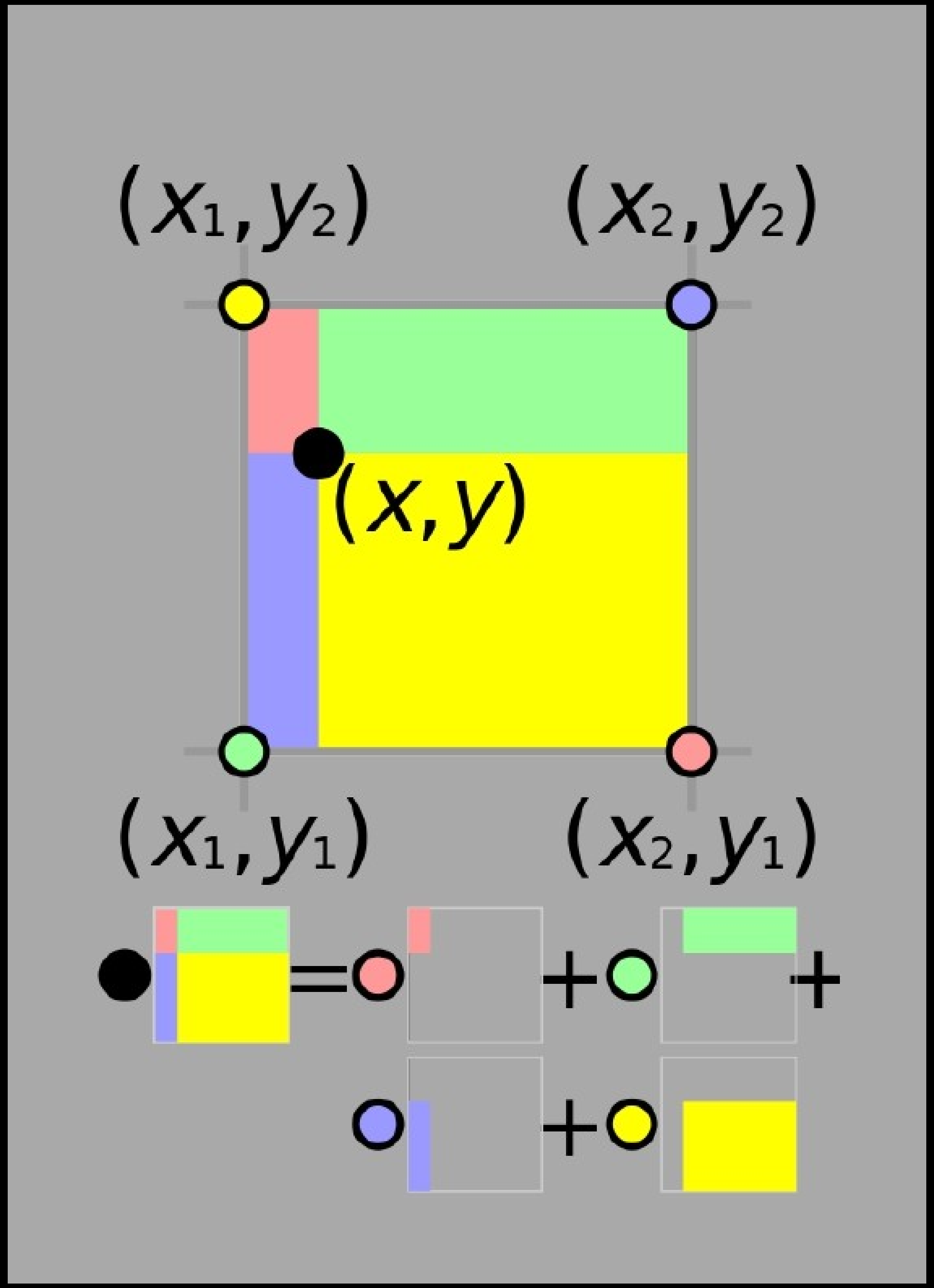






[illegible] [illegible]

[illegible]



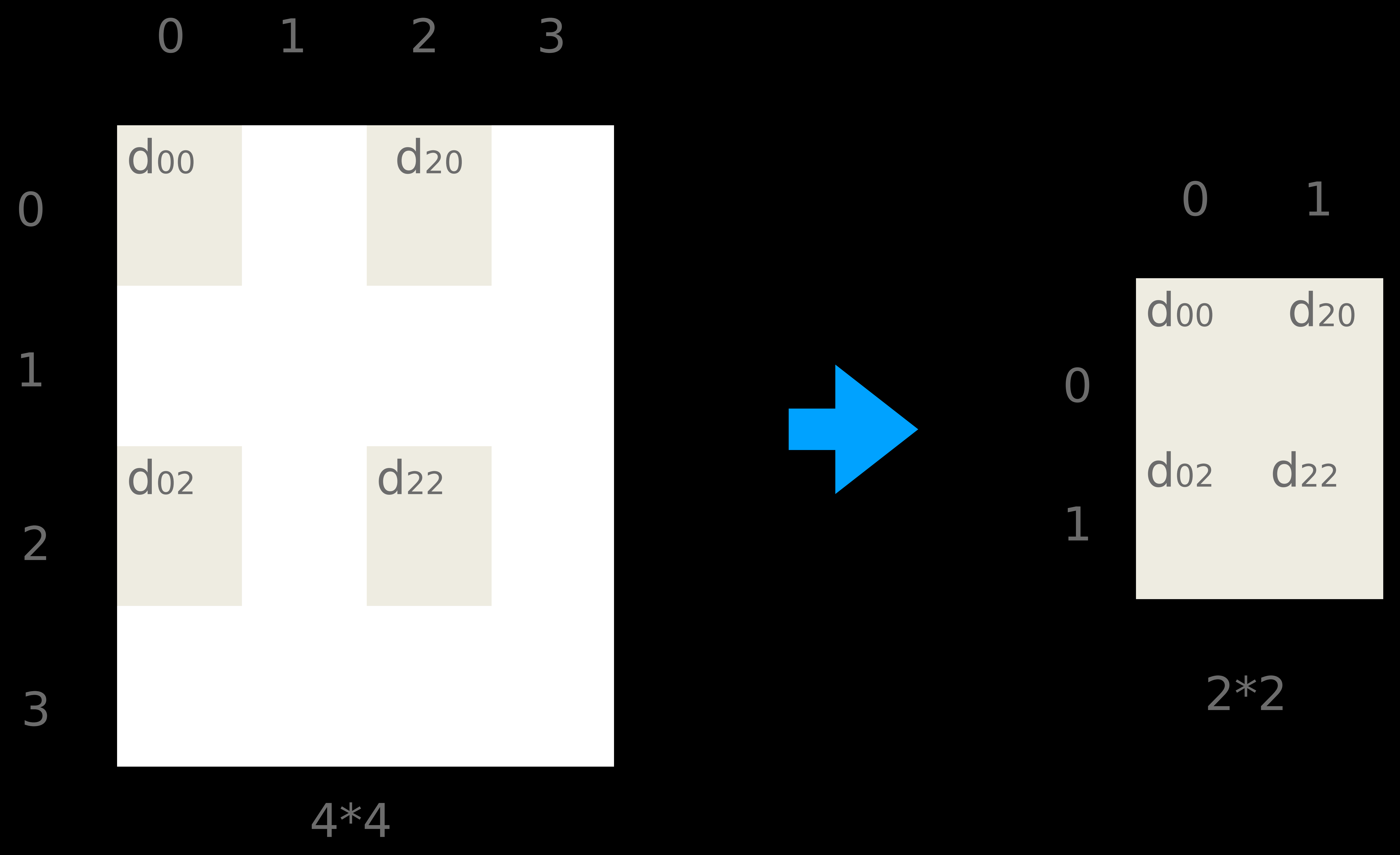
[illegible]

Value at (x,y) =
Sum of the value at each spot multiplied
by the area of the rectangle divided by the
total area of all four rectangles

https://en.wikipedia.org/wiki/Bilinear_interpolation

$$f(i+u,j+v) = (1-u)(1-v)f(i,j) + (1-u)vf(i,j+1) + u(1-v)f(i+1,j) + uvf(i+1,j+1)$$

□□□□□□□□□□



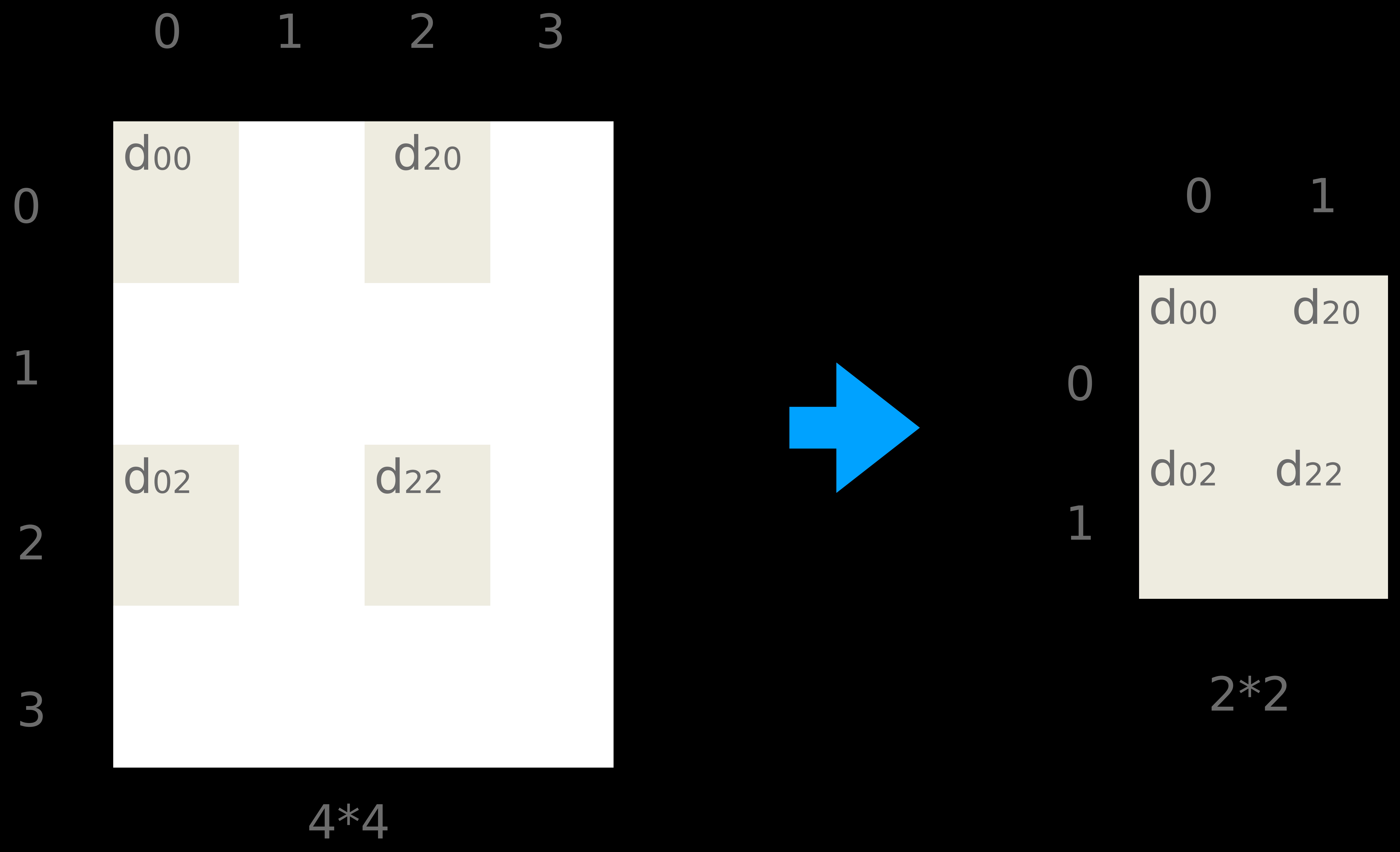
	0	1	2	3
0	d_{00}		d_{20}	
1				
2	d_{02}		d_{22}	
3				

4*4

	0	1
0	d_{00}	d_{20}
1	d_{02}	d_{22}

2*2

□□□□□□□“□□”



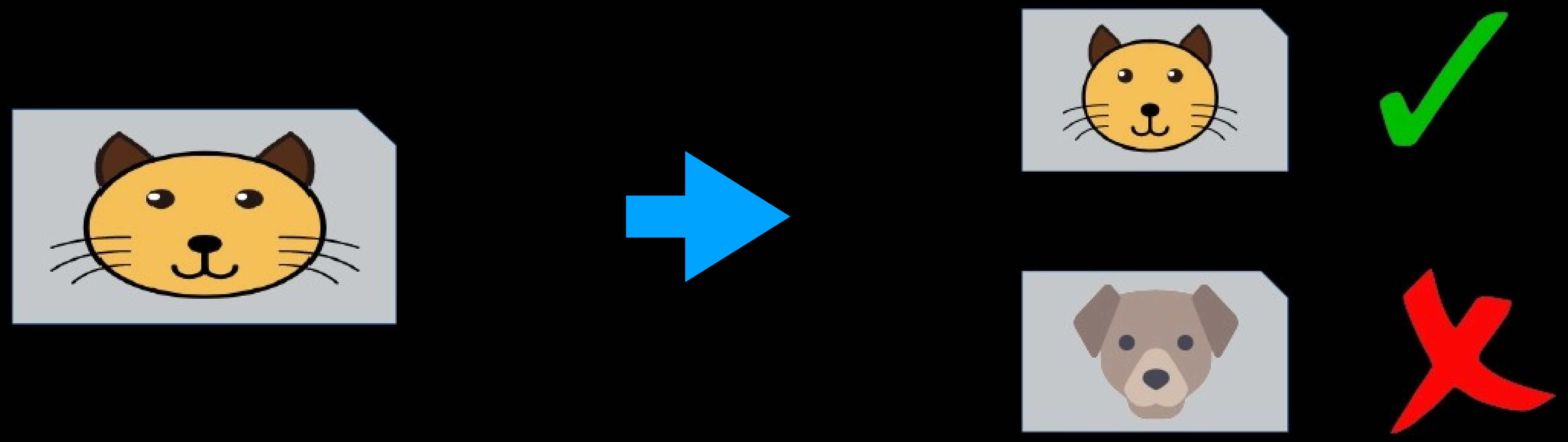
	0	1	2	3
0	d_{00}		d_{20}	
1				
2	d_{02}		d_{22}	
3				

4*4

	0	1
0	d_{00}	d_{20}
1	d_{02}	d_{22}

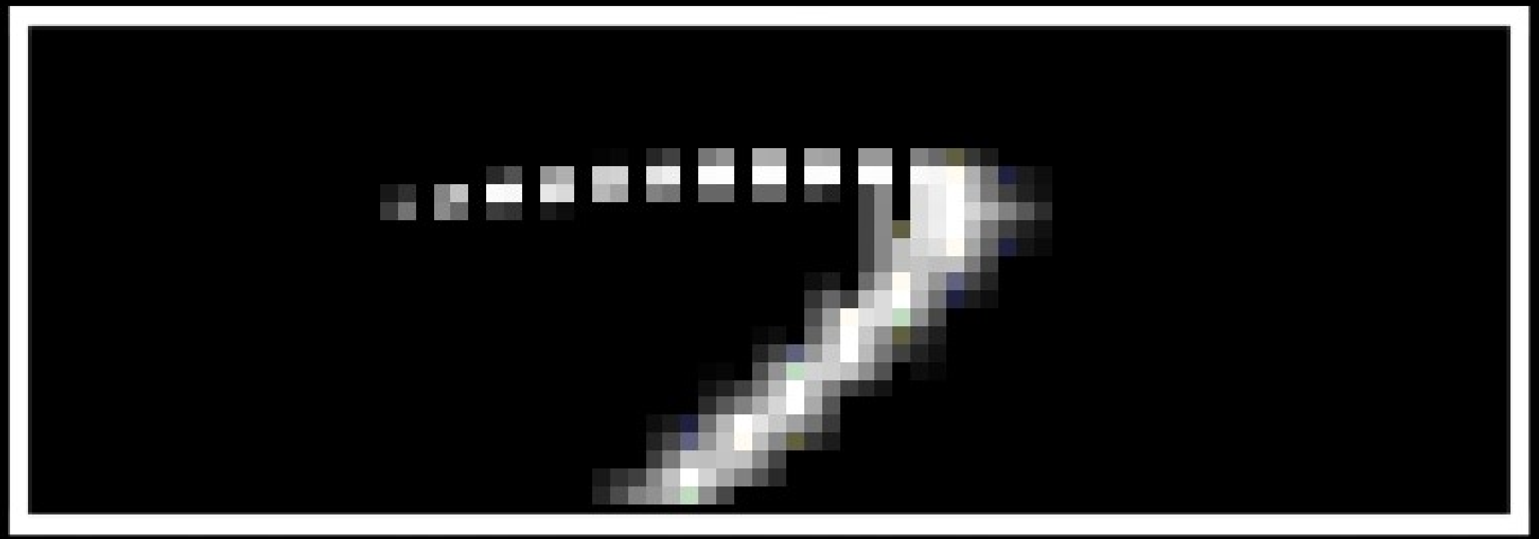
2*2

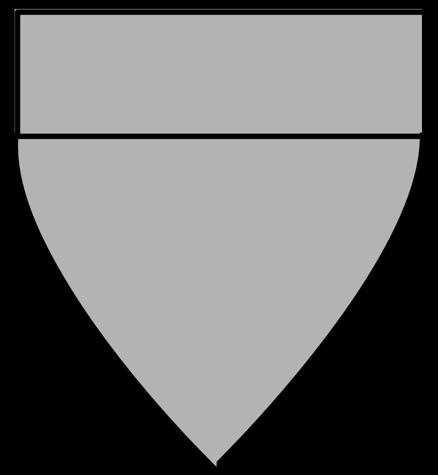
[illegible]



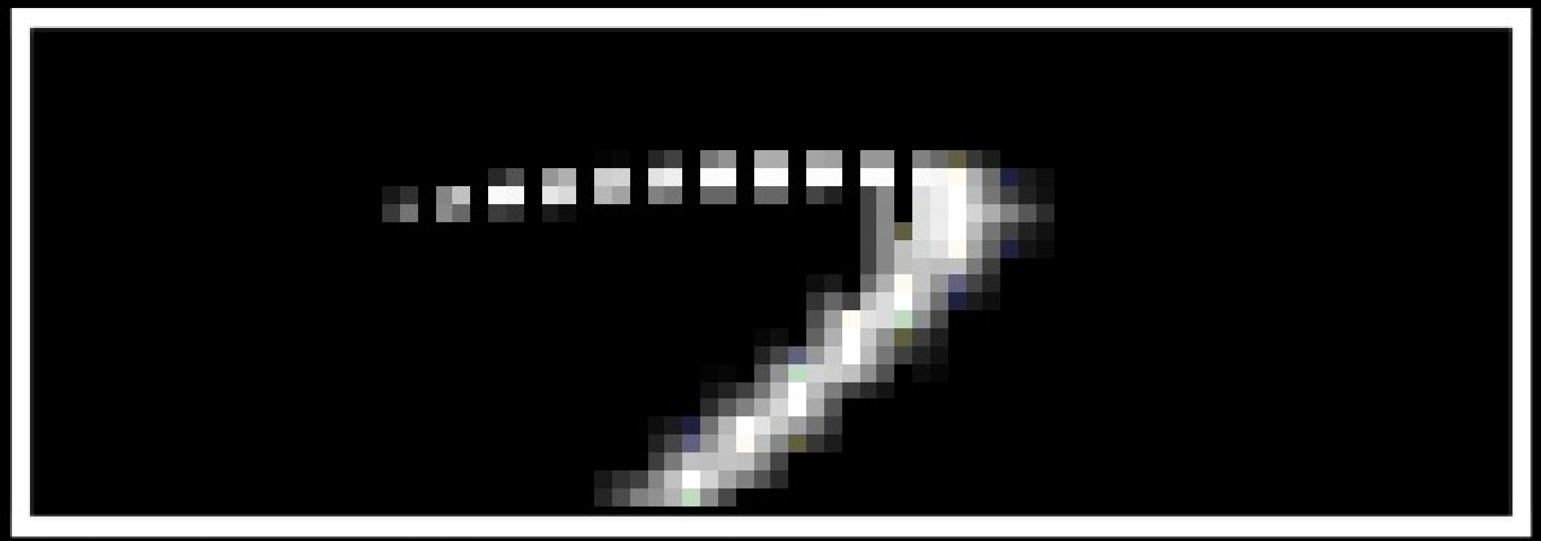
[illegible]“[illegible]”[illegible]

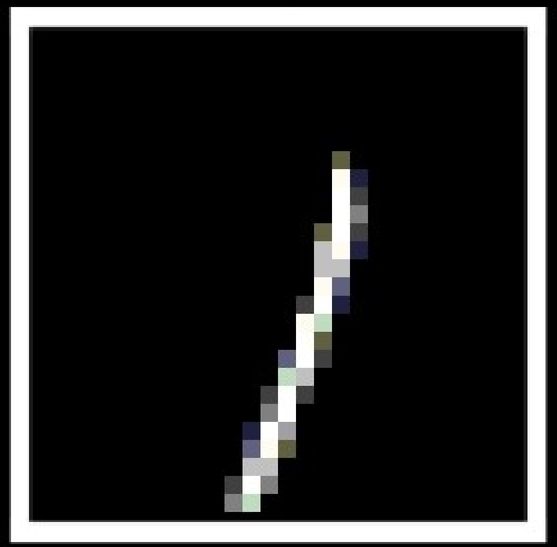
□□□□□□

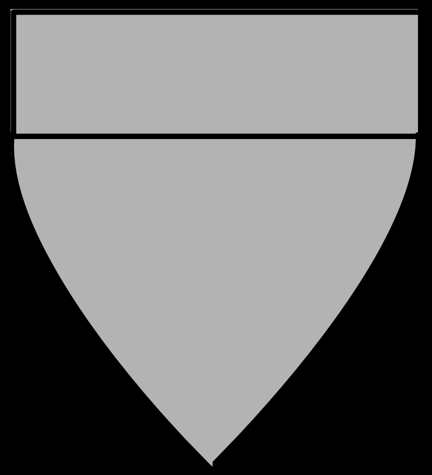




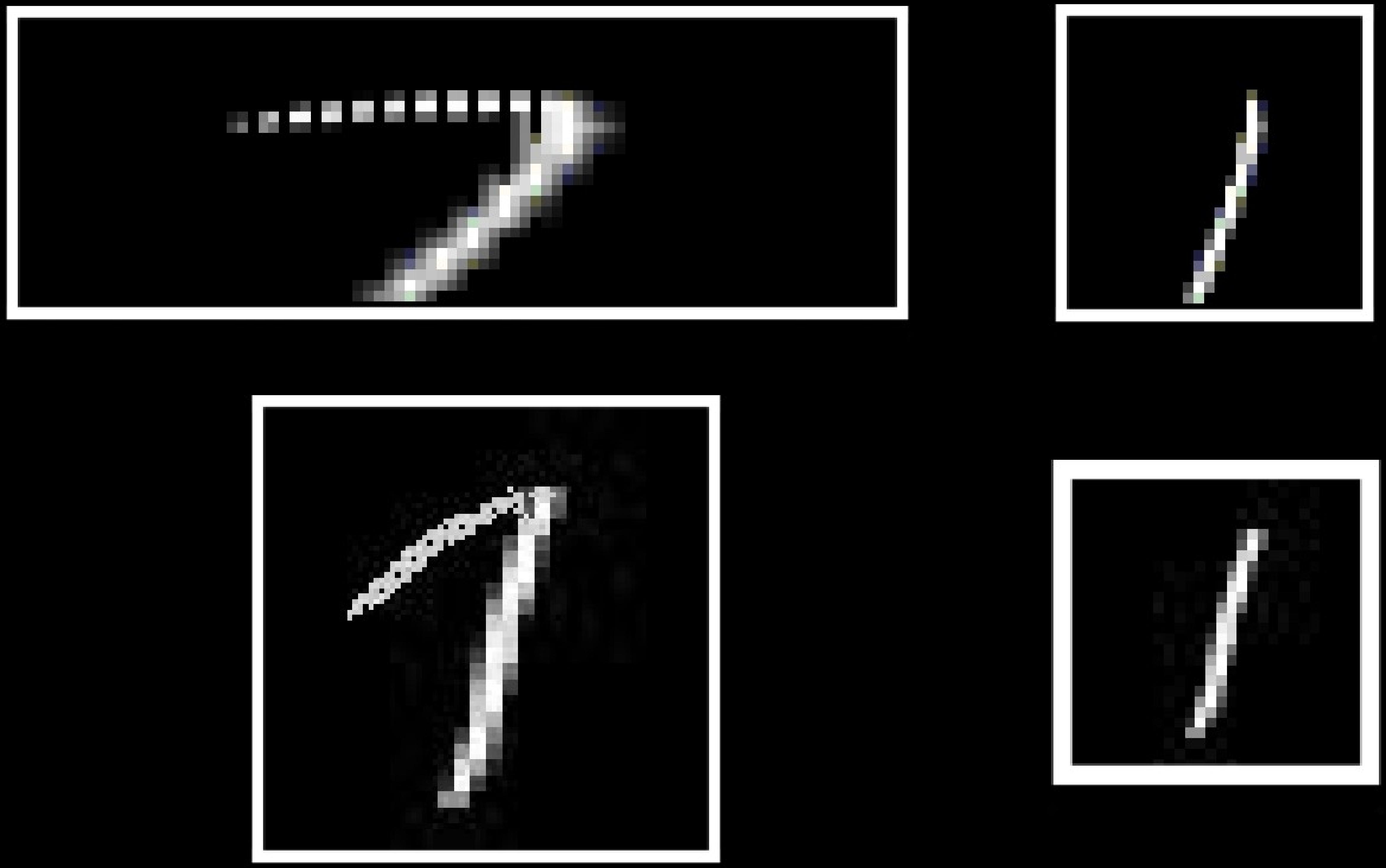
[illegible]







□□□□□□

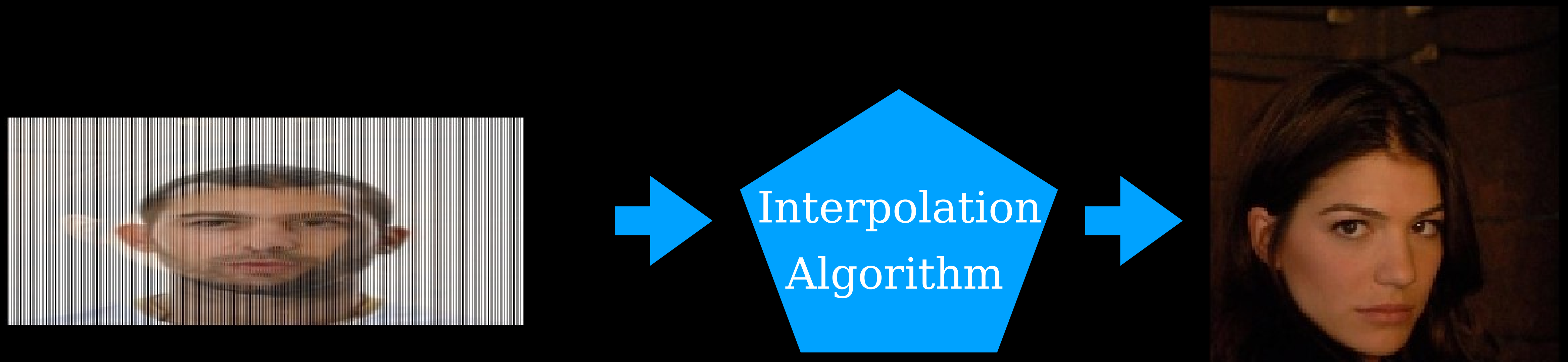


[illegible]

[illegible]

[illegible]

An image is a matrix,
and scaling is done
with convert matrix

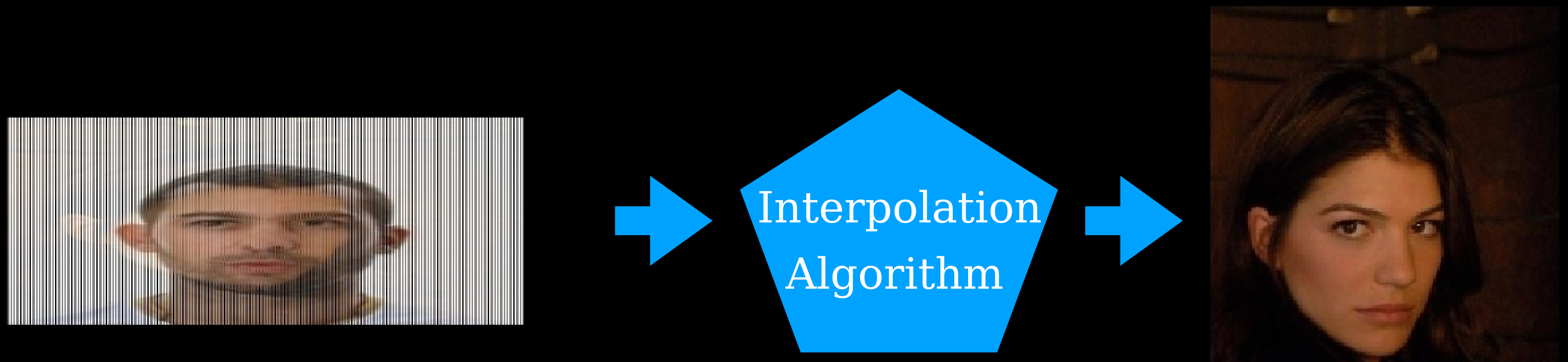


Attack Image
(matrix A)

Target Image
(matrix T)

[illegible]

An image is a matrix, and scaling is done with convert matrix



Interpolation Algorithm

Attack Image

(matrix A)

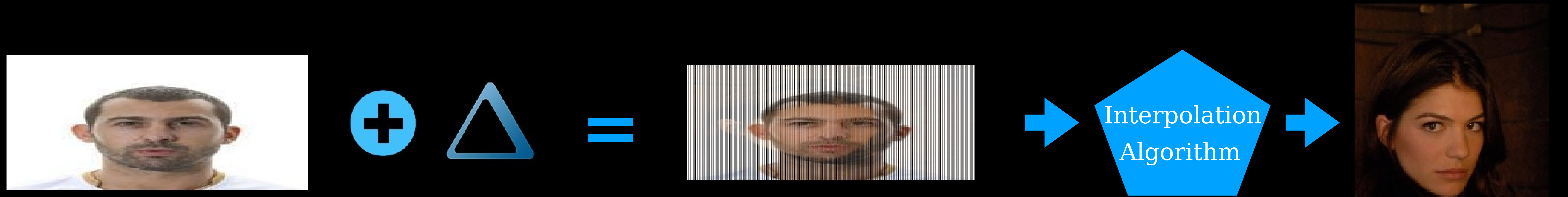
Target Image

(matrix T)

$$C\,[p,m] = \begin{bmatrix} C\_1,1 & -- & C\_1,m \\ \vdots & \ddots & \\ C\_p,1 & & C\_p,m \end{bmatrix}$$

$$C[p,m] \times A[m,n] \times C[n,q] = T[p,q]$$

[illegible]



Interpolation Algorithm

Input Image (matrix S)

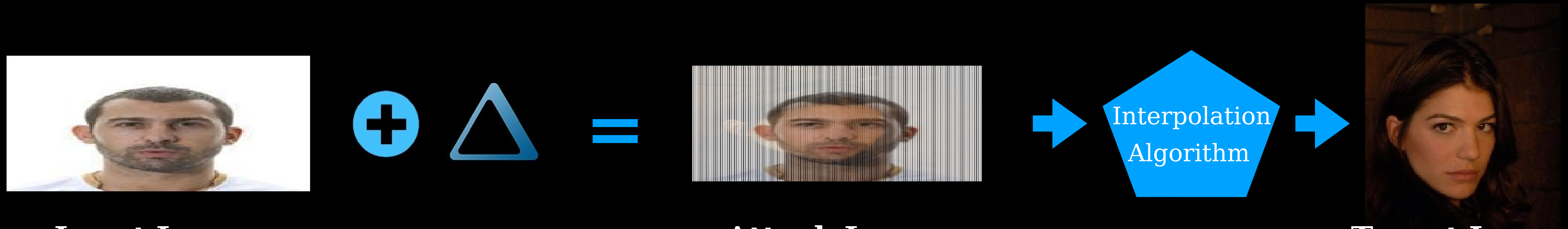
Attack Image (matrix A)

Target Image (matrix T)

$$S[\,]+\Delta[\,]=A[\,]$$

$$A[\,]\times C[\,]=T[\,]$$

[illegible]



Input Image (matrix S)

Attack Image (matrix A)

Interpolation Algorithm

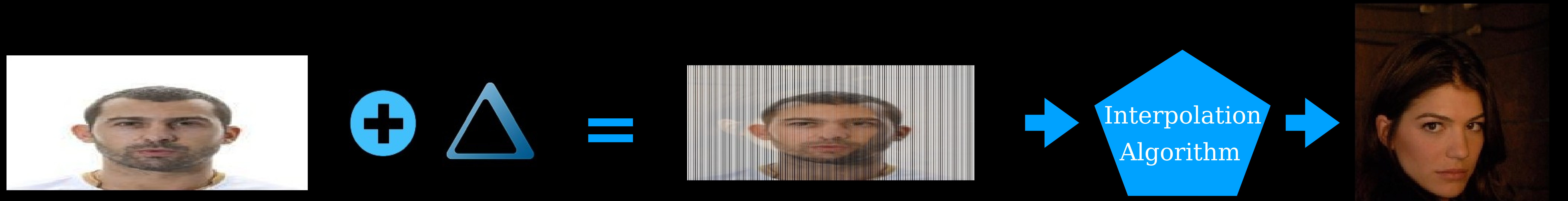
Target Image (matrix T)

$$S[] + \Delta [] = A[]$$

$$A[] \text{x} C [] = T[]$$

[illegible]

[illegible]



Interpolation Algorithm

Input Image (matrix S)

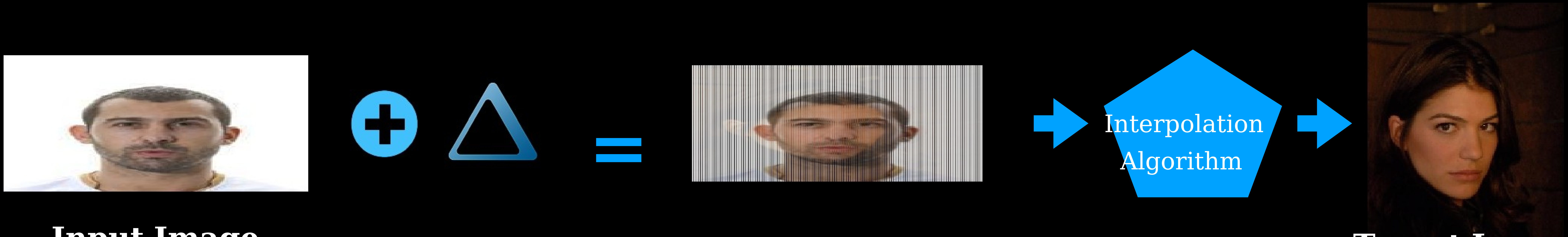
Attack Image (matrix A)

Target Image (matrix T)

$$S[\,] + \Delta[\,] = A[\,]$$

$$A[\,] \times C[\,] = T[\,]$$

[illegible]



Input Image
(matrix S)

Attack Image
(matrix A)

Interpolation
Algorithm

Target Image
(matrix T)

$$S[] + \Delta [] = A[]$$

$$A[] \times C [] = T[]$$

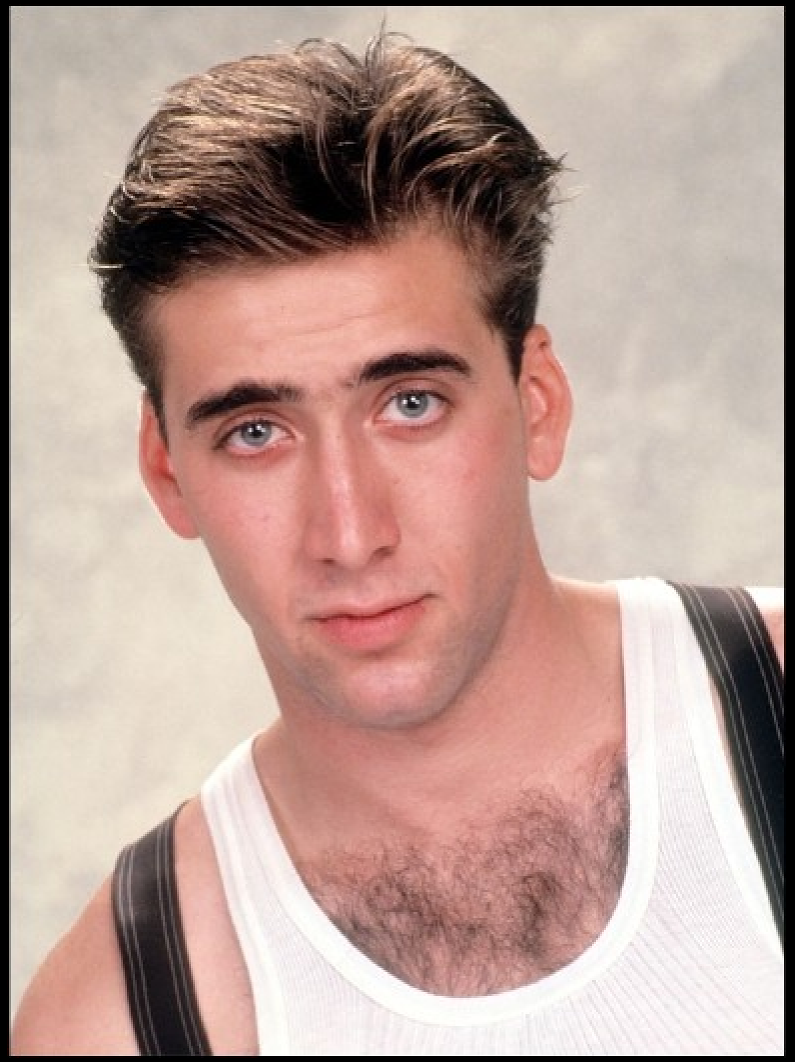
[illegible]

[illegible]

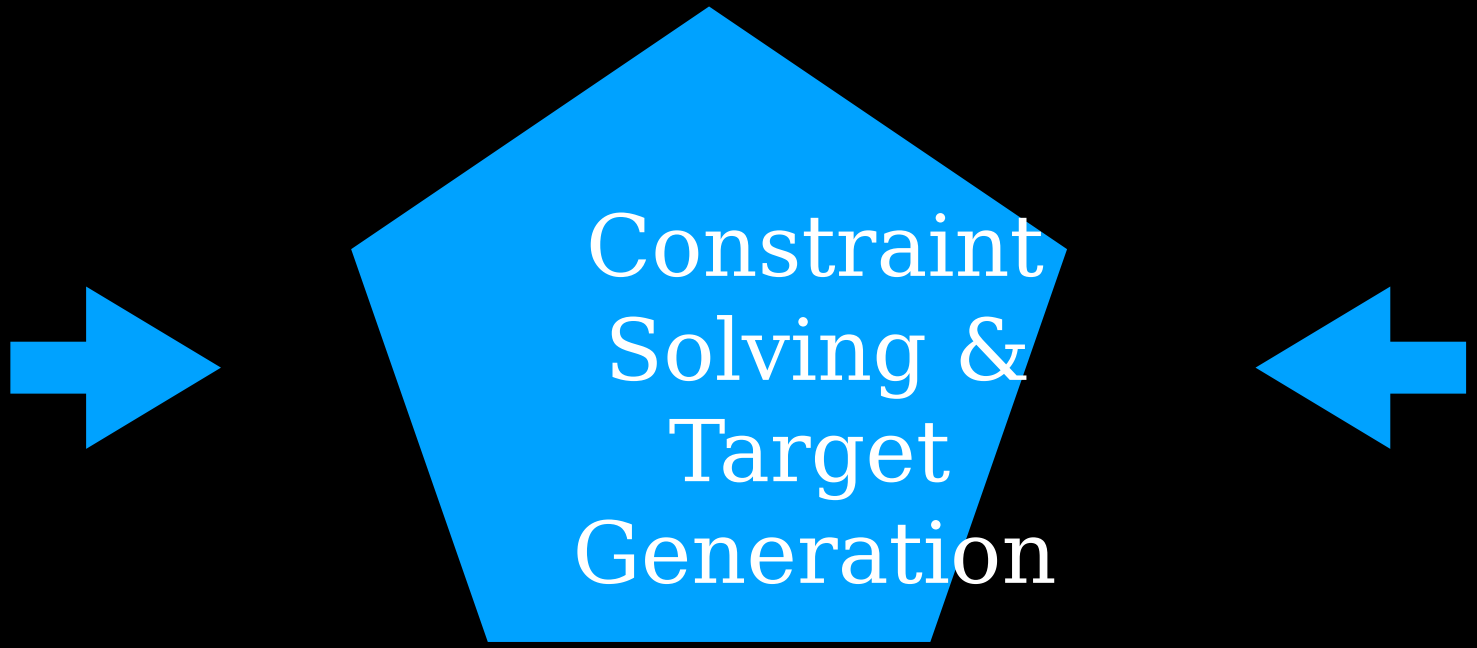
[illegible] 300 [illegible]

[illegible] …[illegible]

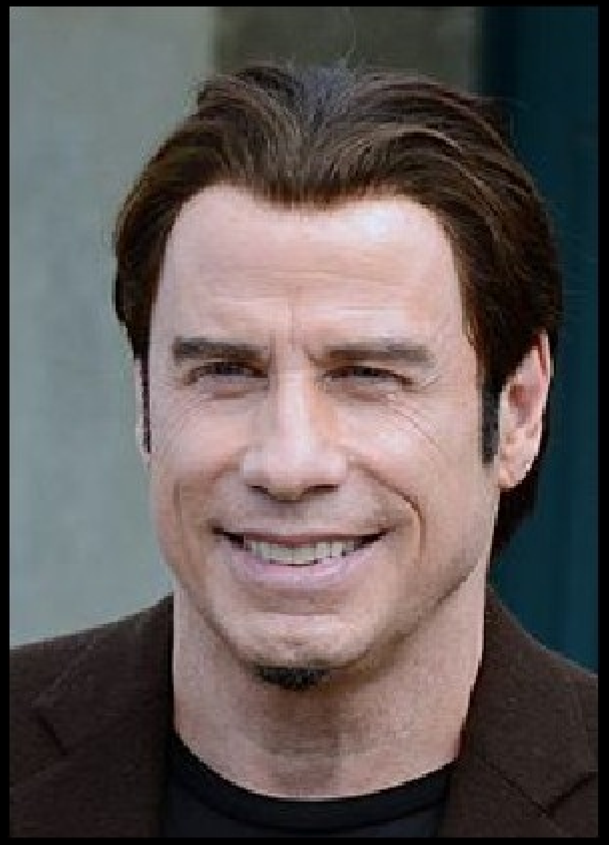
□□□□ #1 (□□)



Specific Image Source

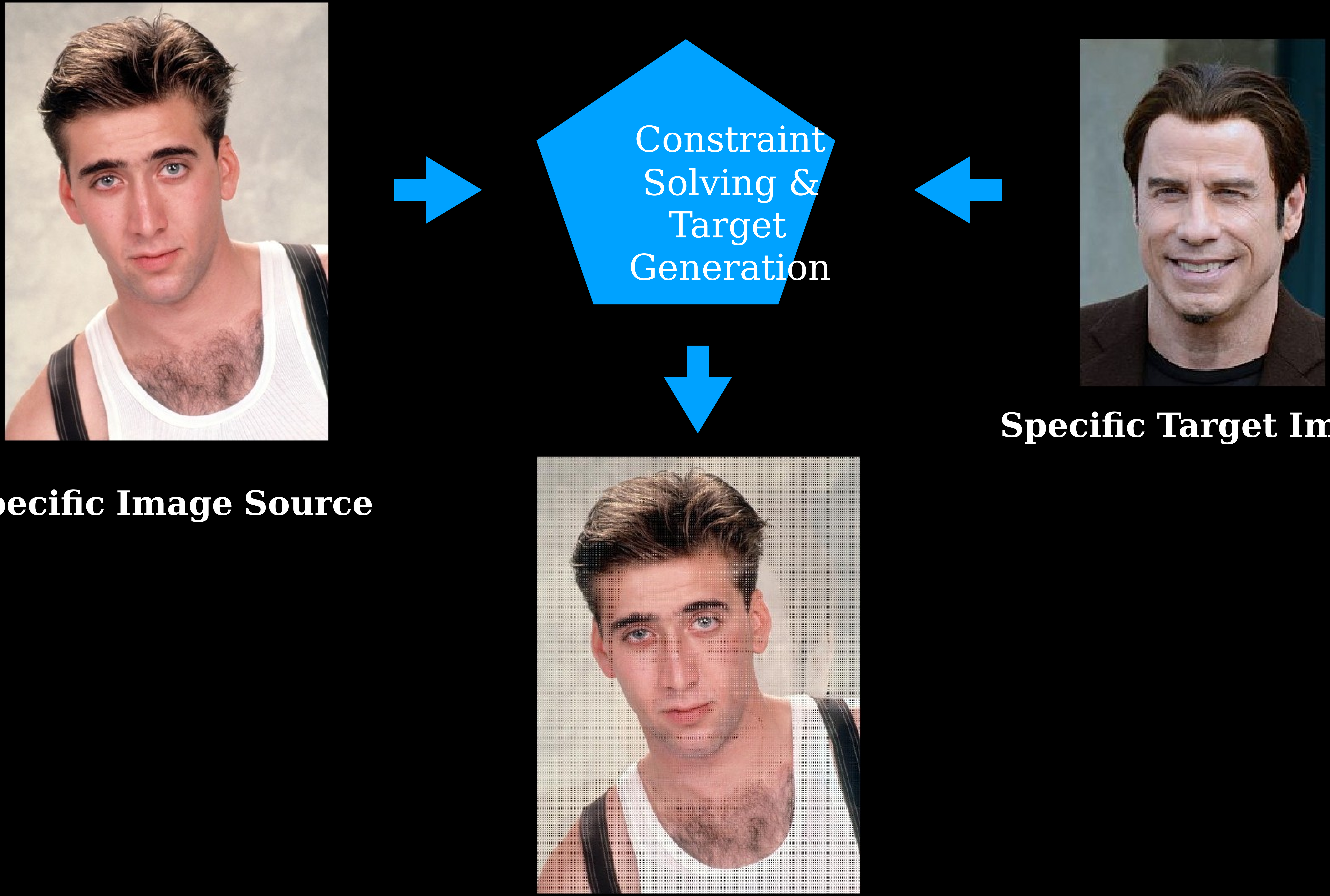


Constraint Solving & Target Generation



Specific Target Image

□□□□ #1 (□□)

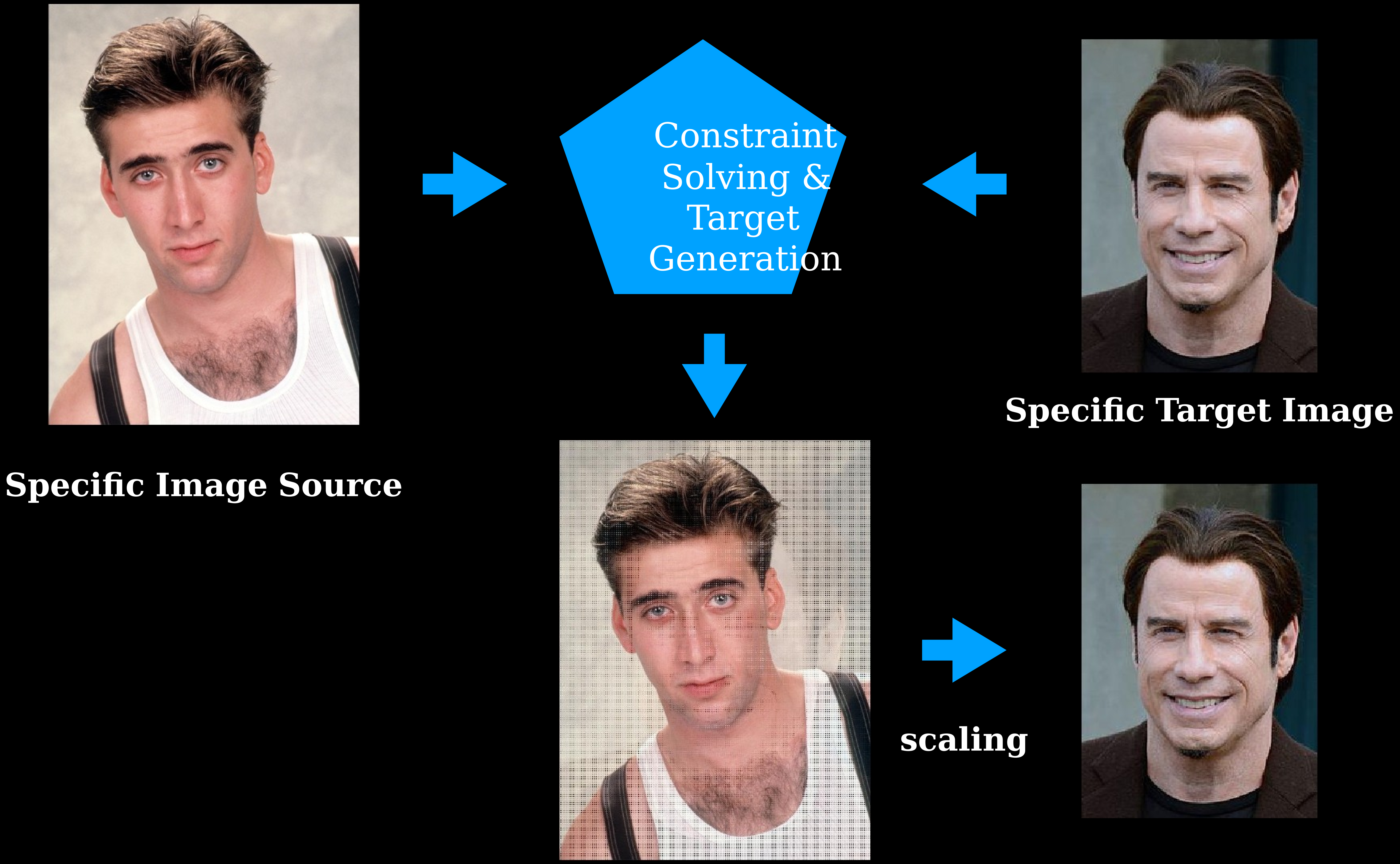


Constraint Solving & Target Generation

Specific Image Source

Specific Target Image

□□□□ #1 (□□)



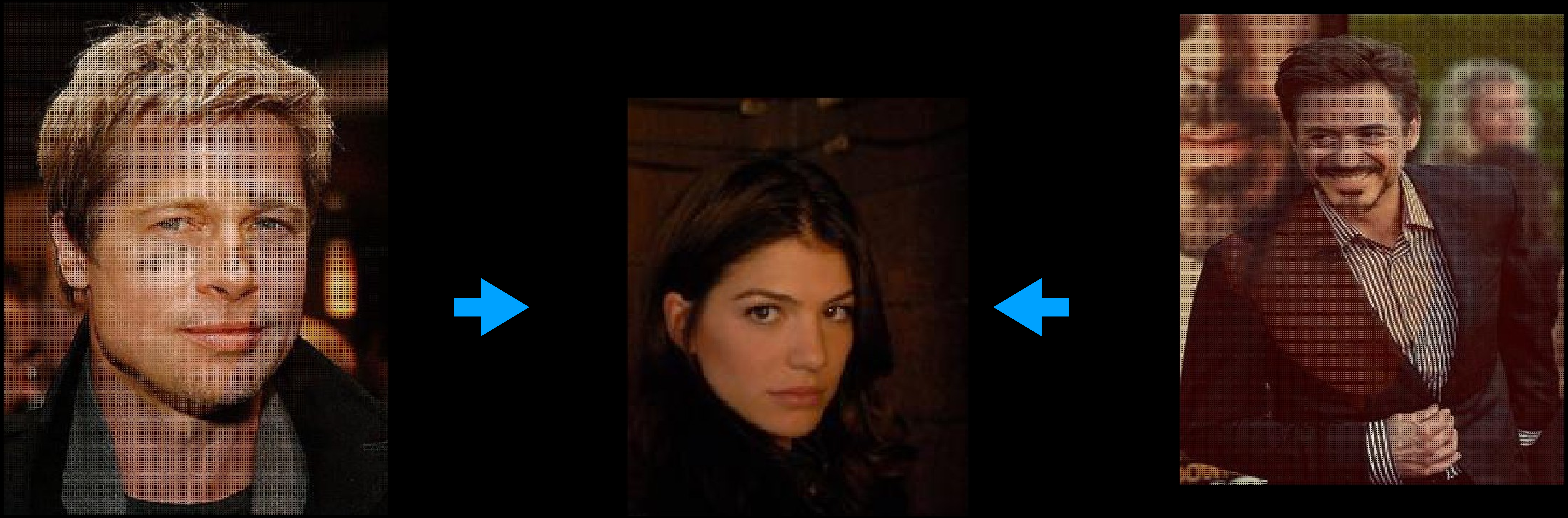
Constraint Solving & Target Generation

Specific Image Source

Specific Target Image

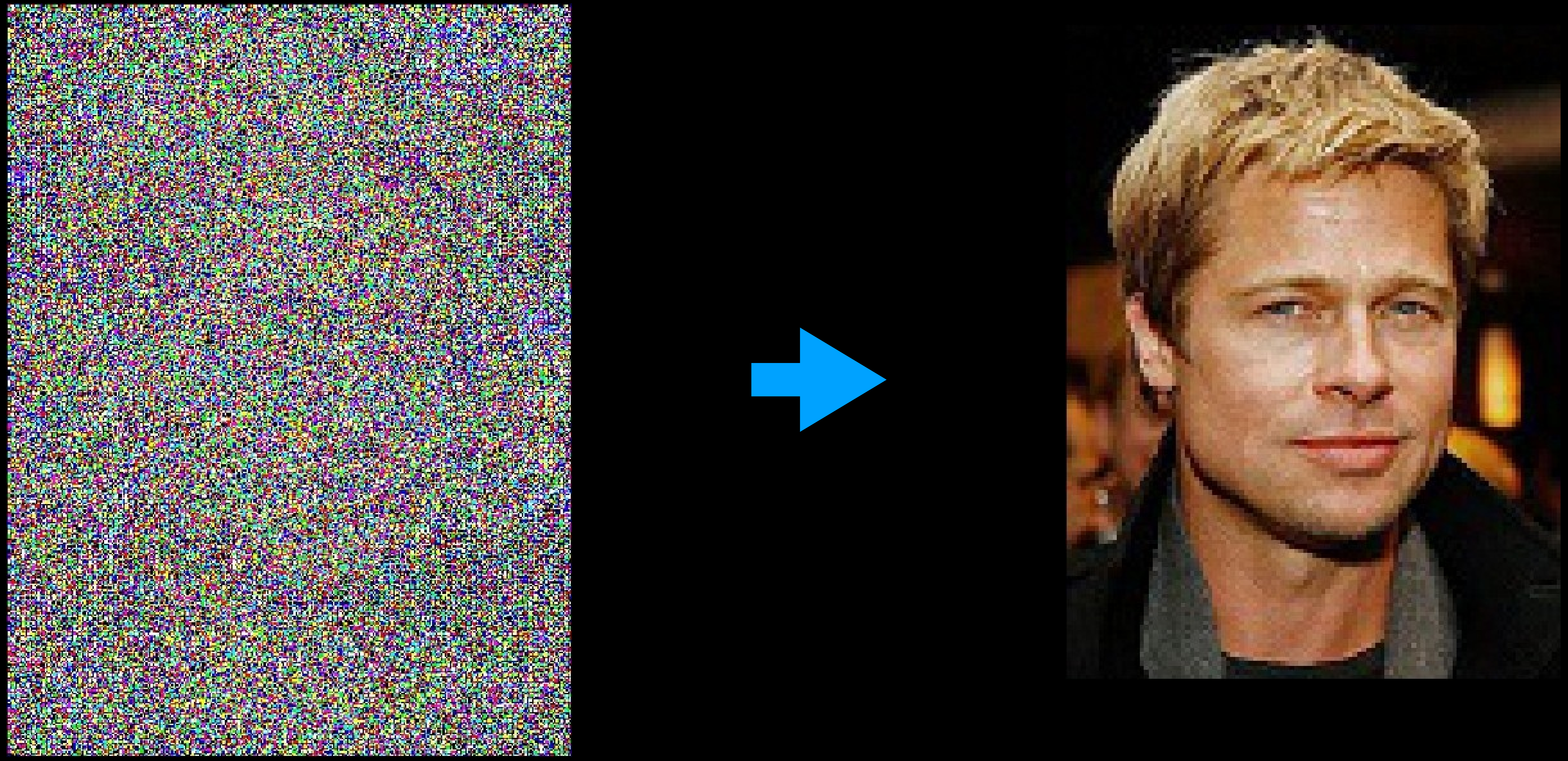
scaling

□□□□ #1 (□□)



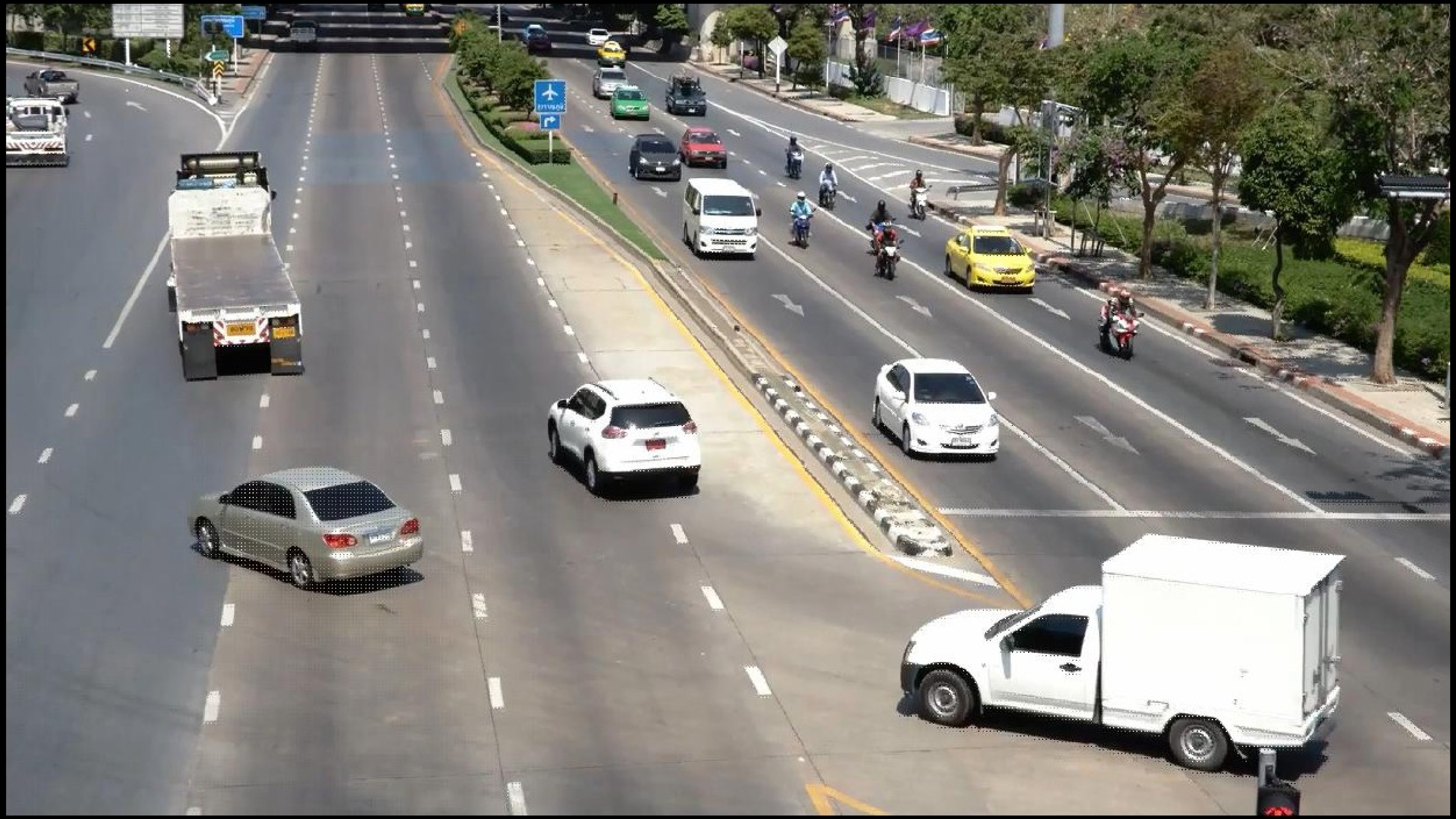
□□□□□□□

[illegible] #1 ([illegible])



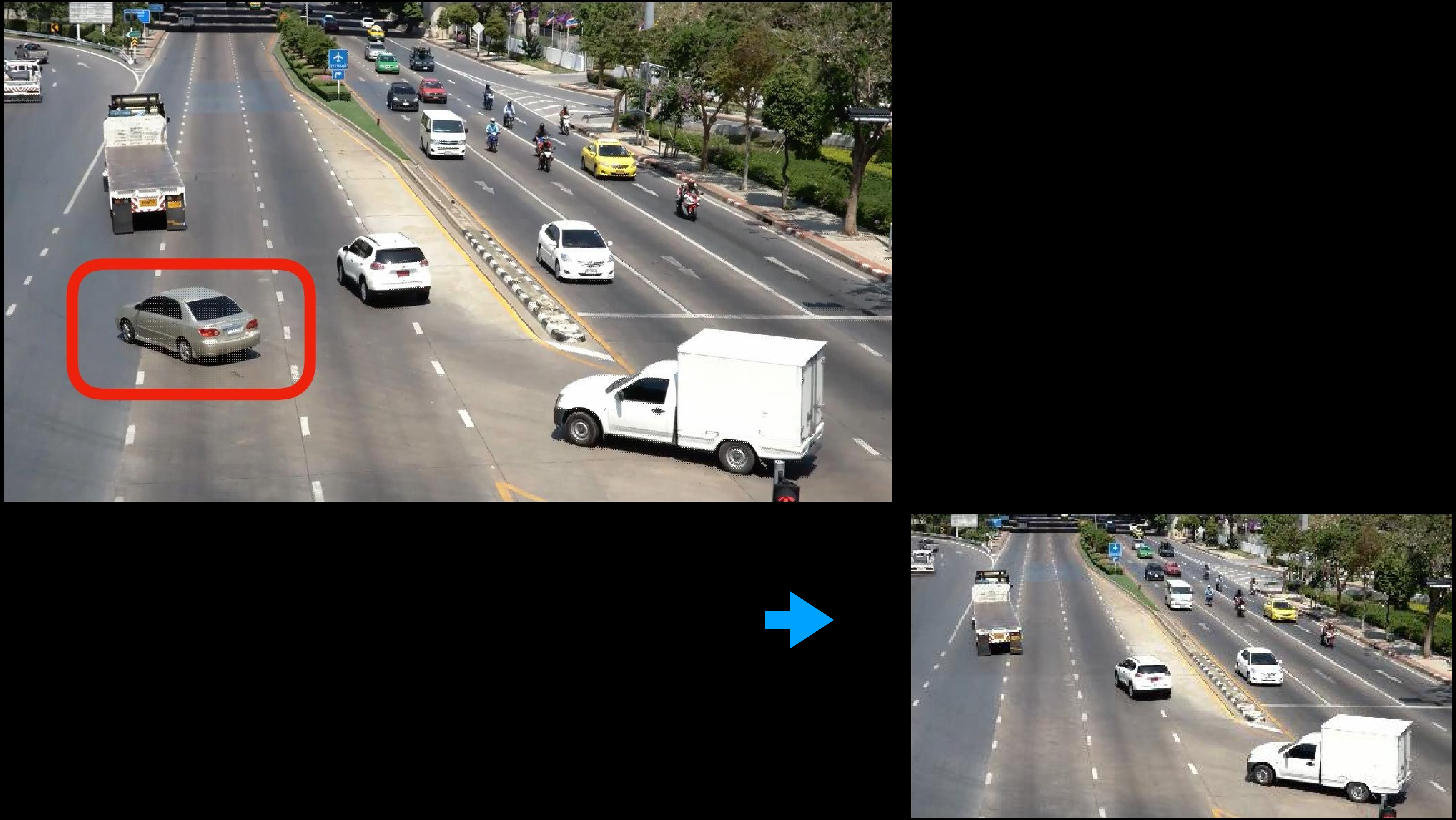
[illegible]

□□□□ #2 (□□□□□)

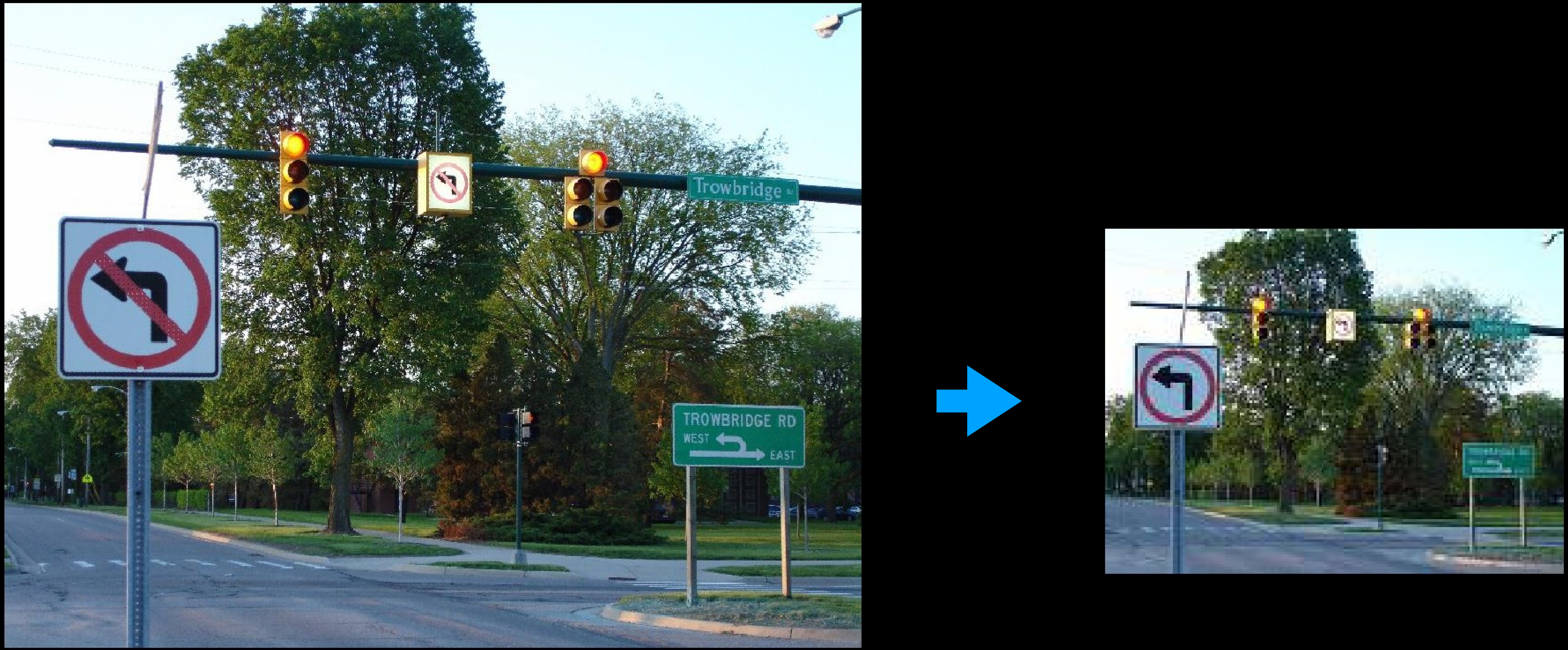


□□□□ #2 (□□□□□)

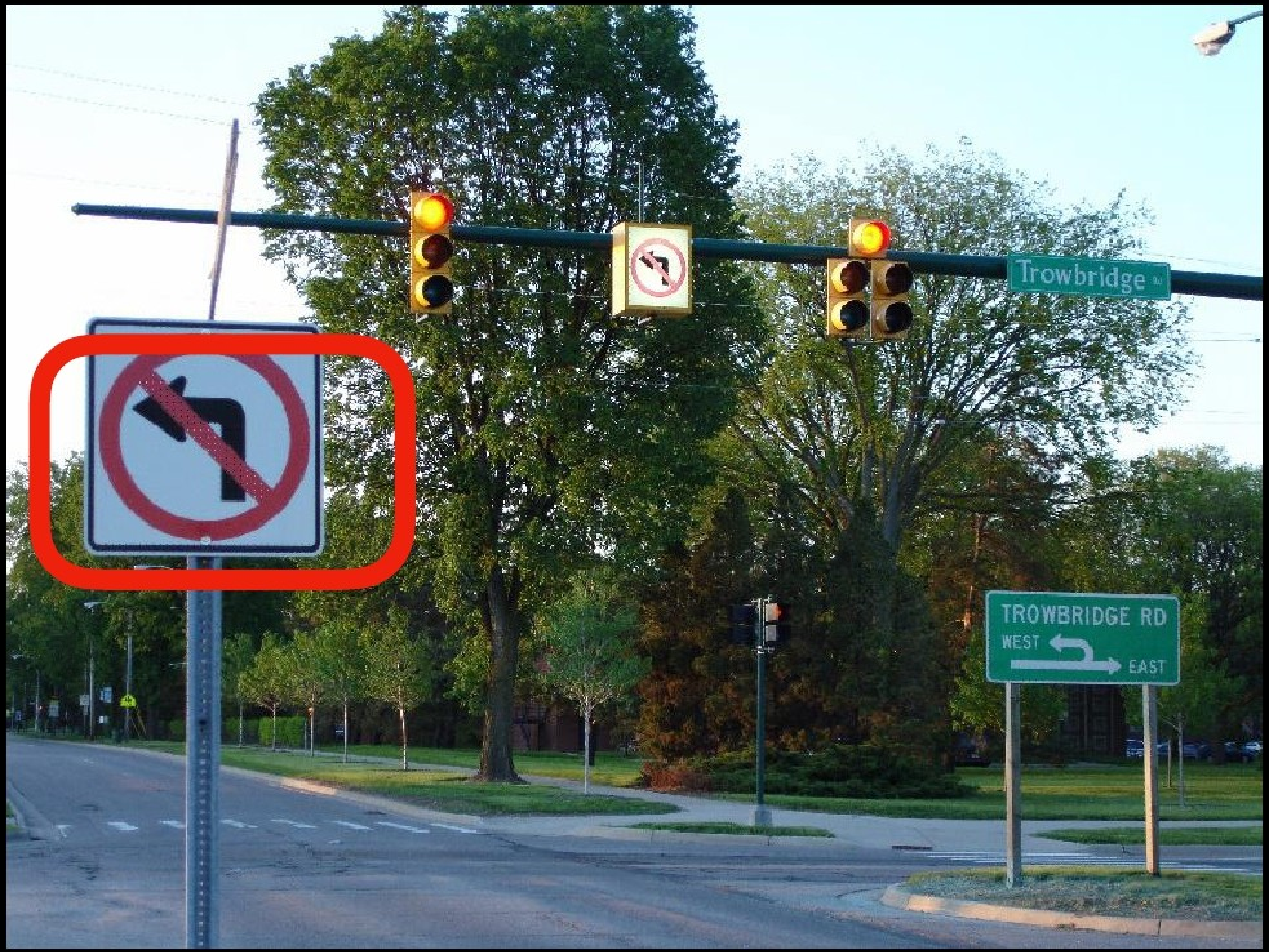
□□□□ #2 (□□□□□)

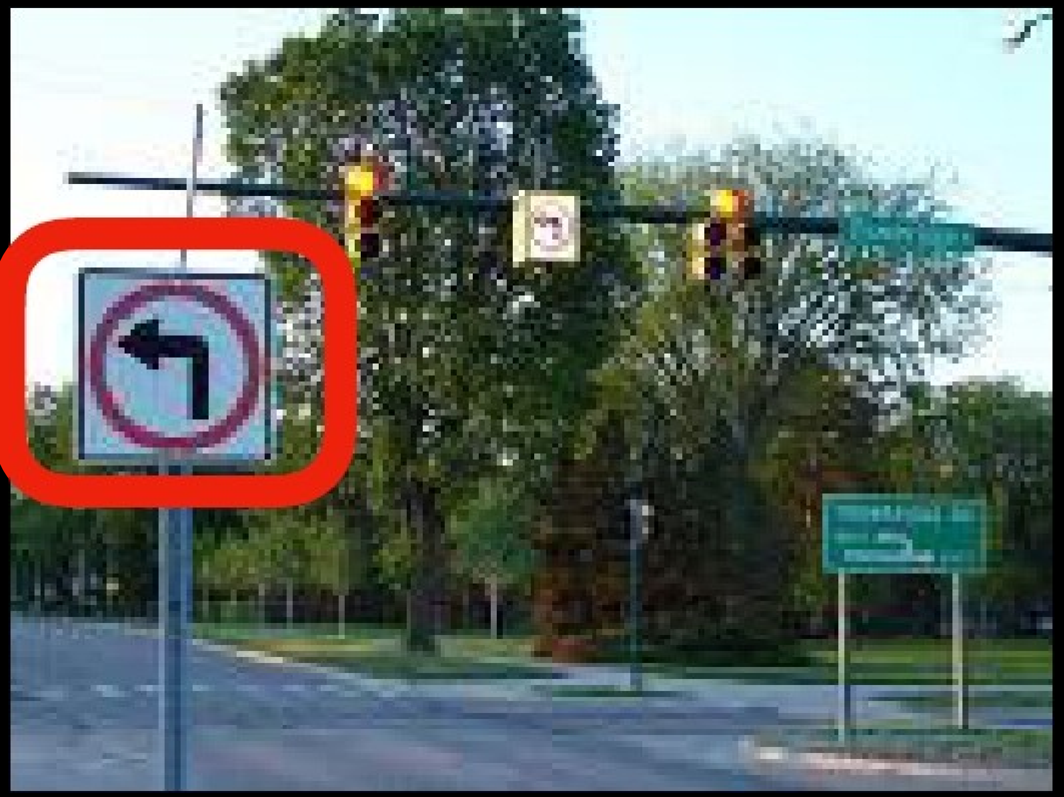


□□□□ #3 (□□□□)

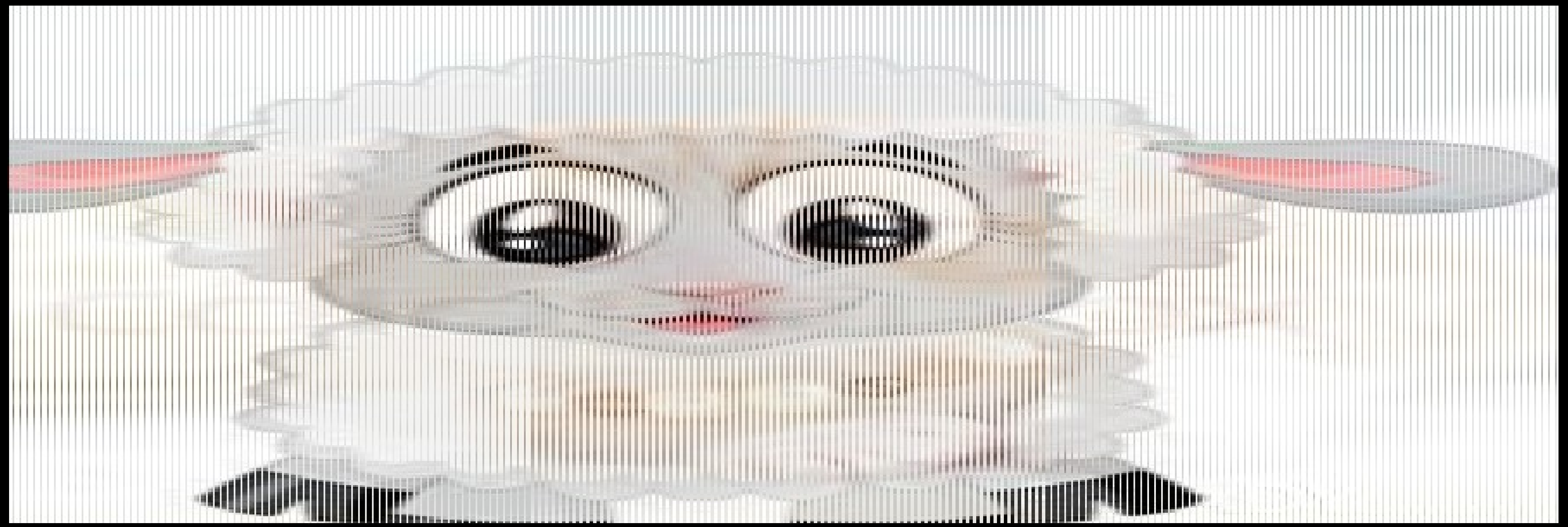


□□□□ #3 (□□□□)

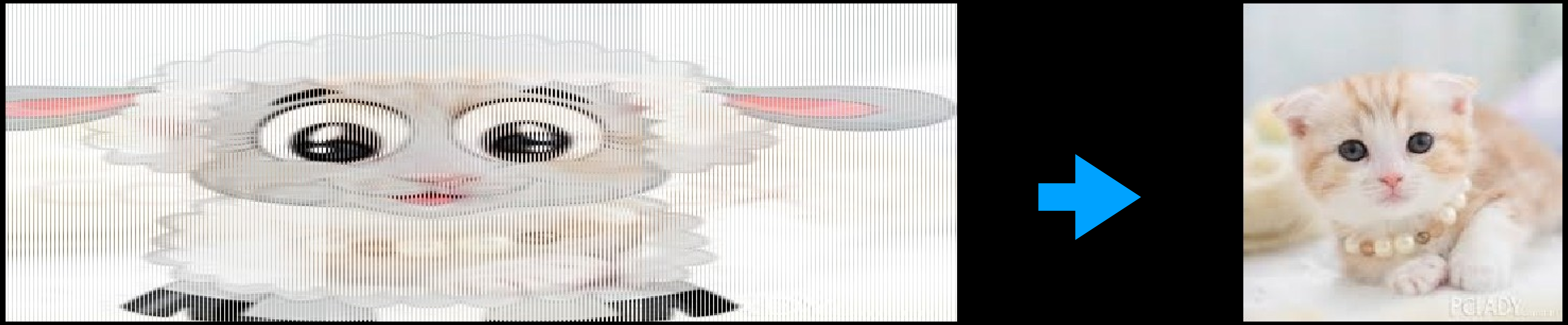




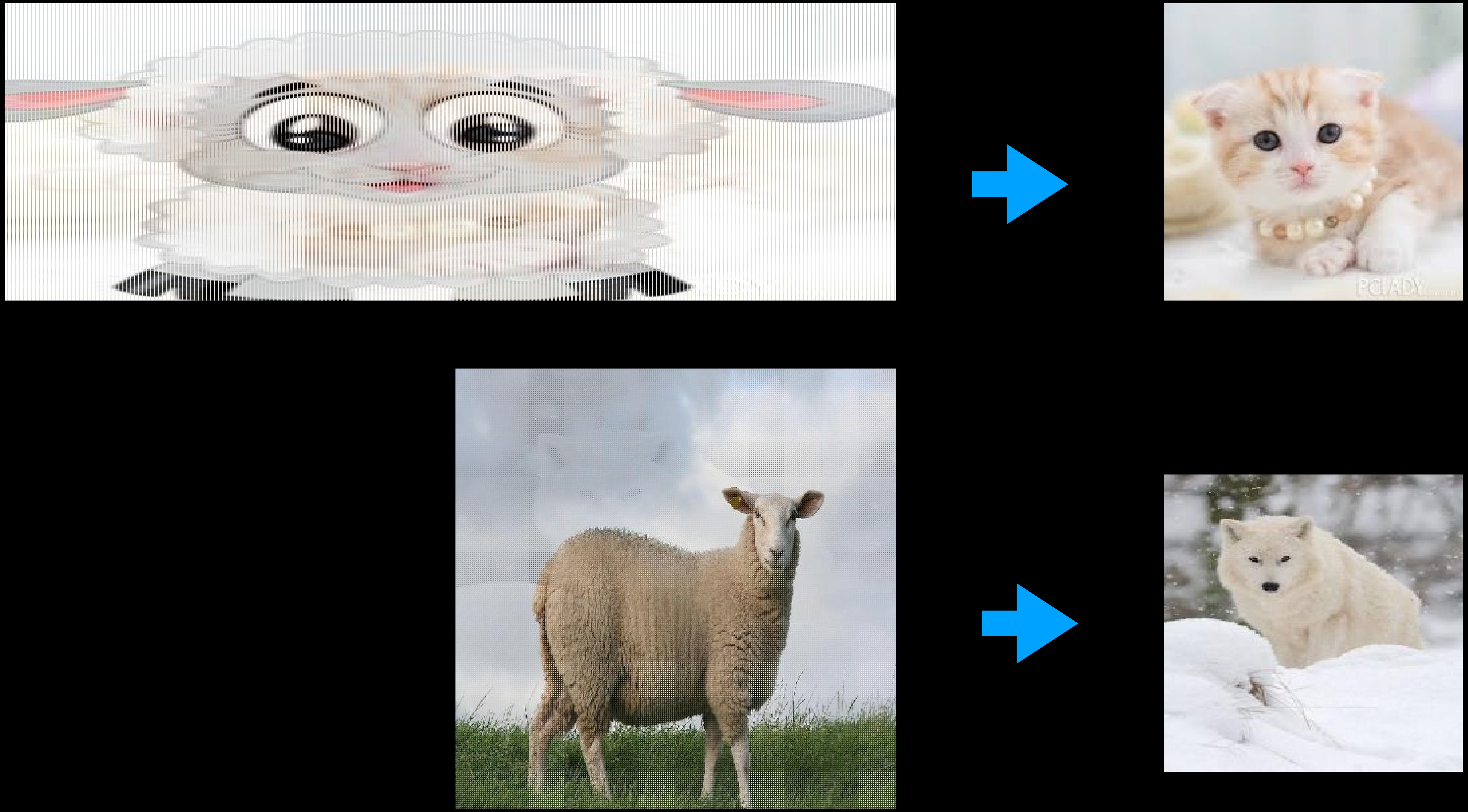
[illegible] #4 ([illegible])



□□□□ #4 (□□□□)



□□□□ #4 (□□□□)



[illegible]

- [illegible]
  - [illegible]
- [illegible]
  - [illegible]
- [illegible]“[illegible]”[illegible]
  - [illegible]“[illegible]”[illegible]

[illegible]“[illegible]”[illegible]

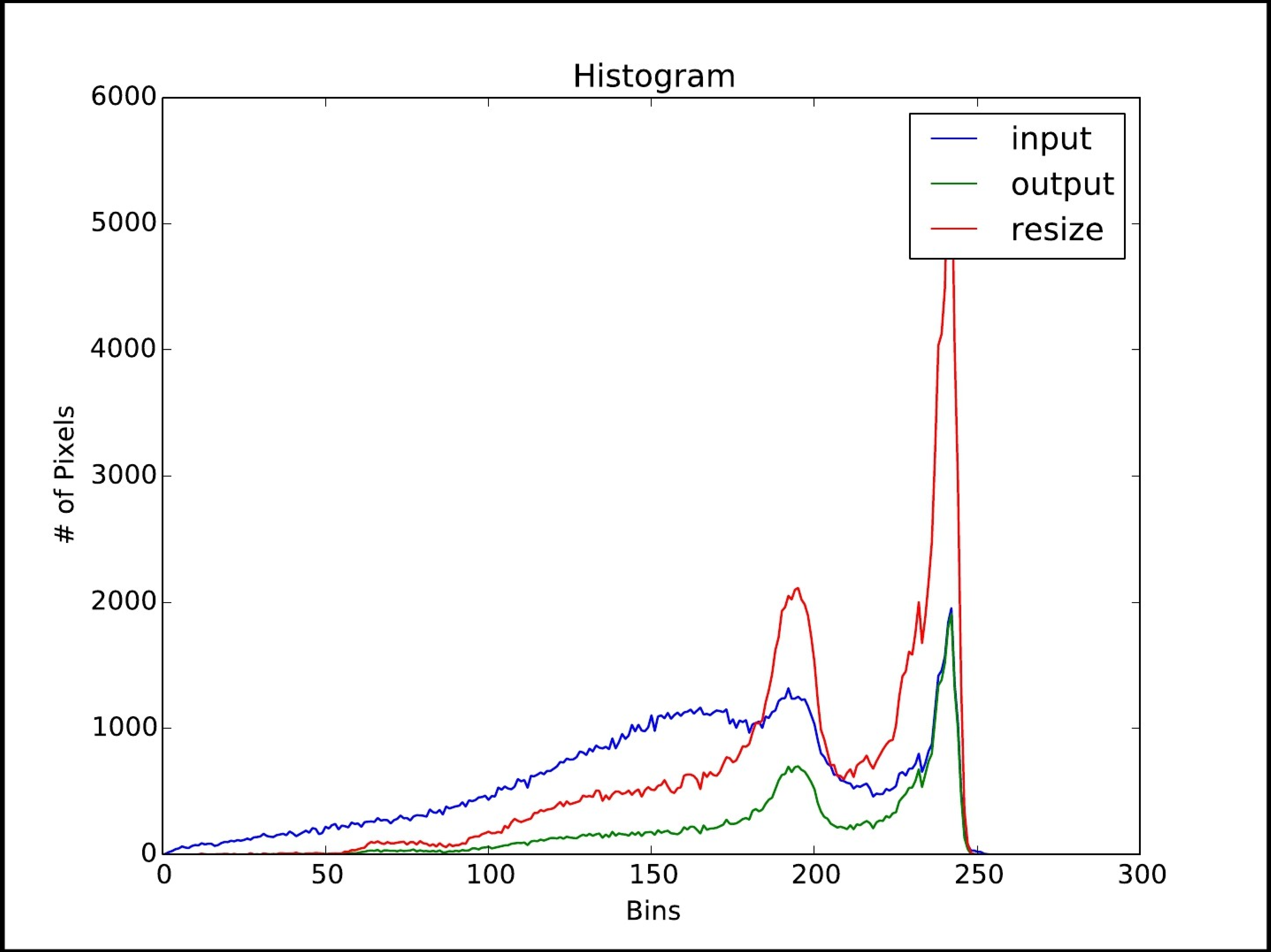
- [illegible]
  - [illegible]
  - Color histogram
  - Color scattering index
- [illegible]
  - [illegible]
  - [illegible]

[illegible]“[illegible]”[illegible]

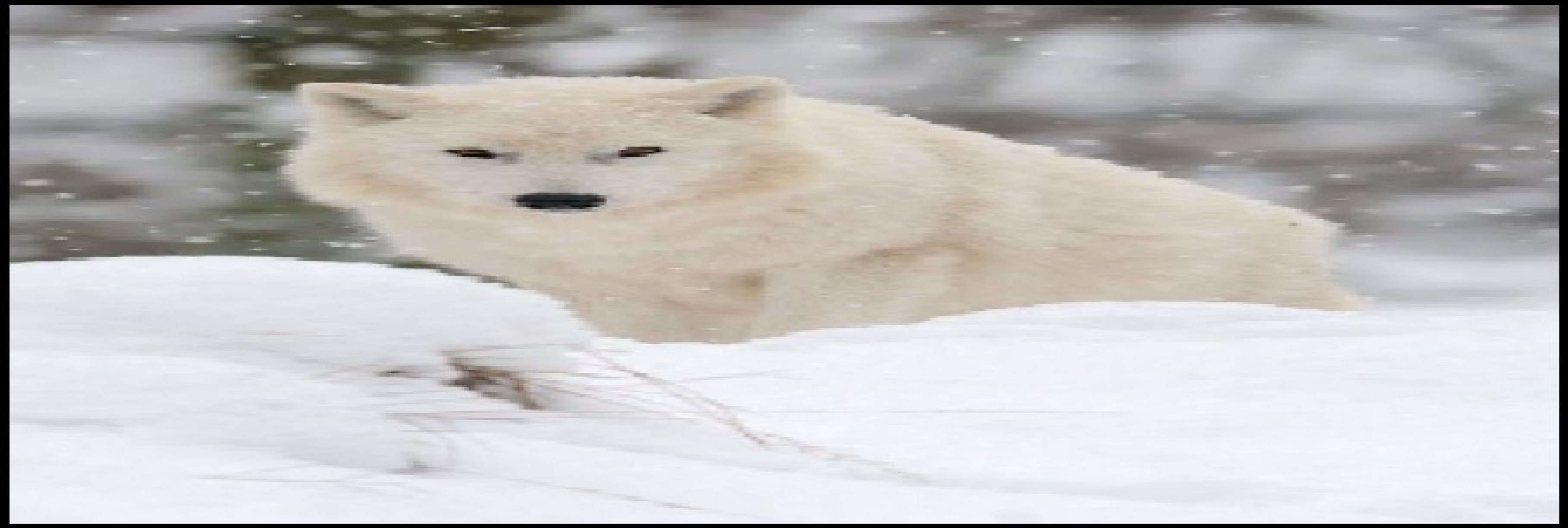
Input

Output

Resize

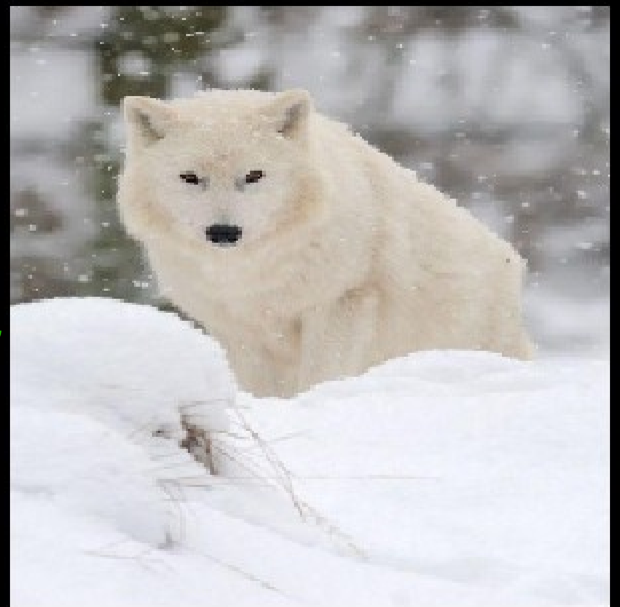


□□□□□□□□□“□□”□□□□

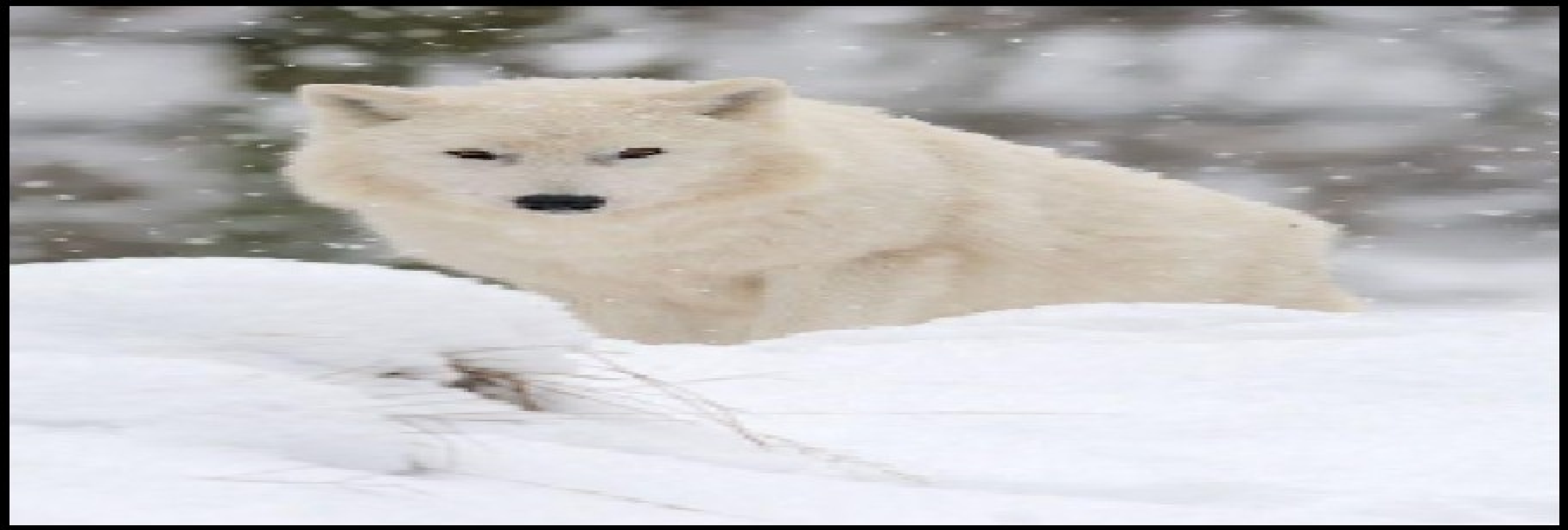


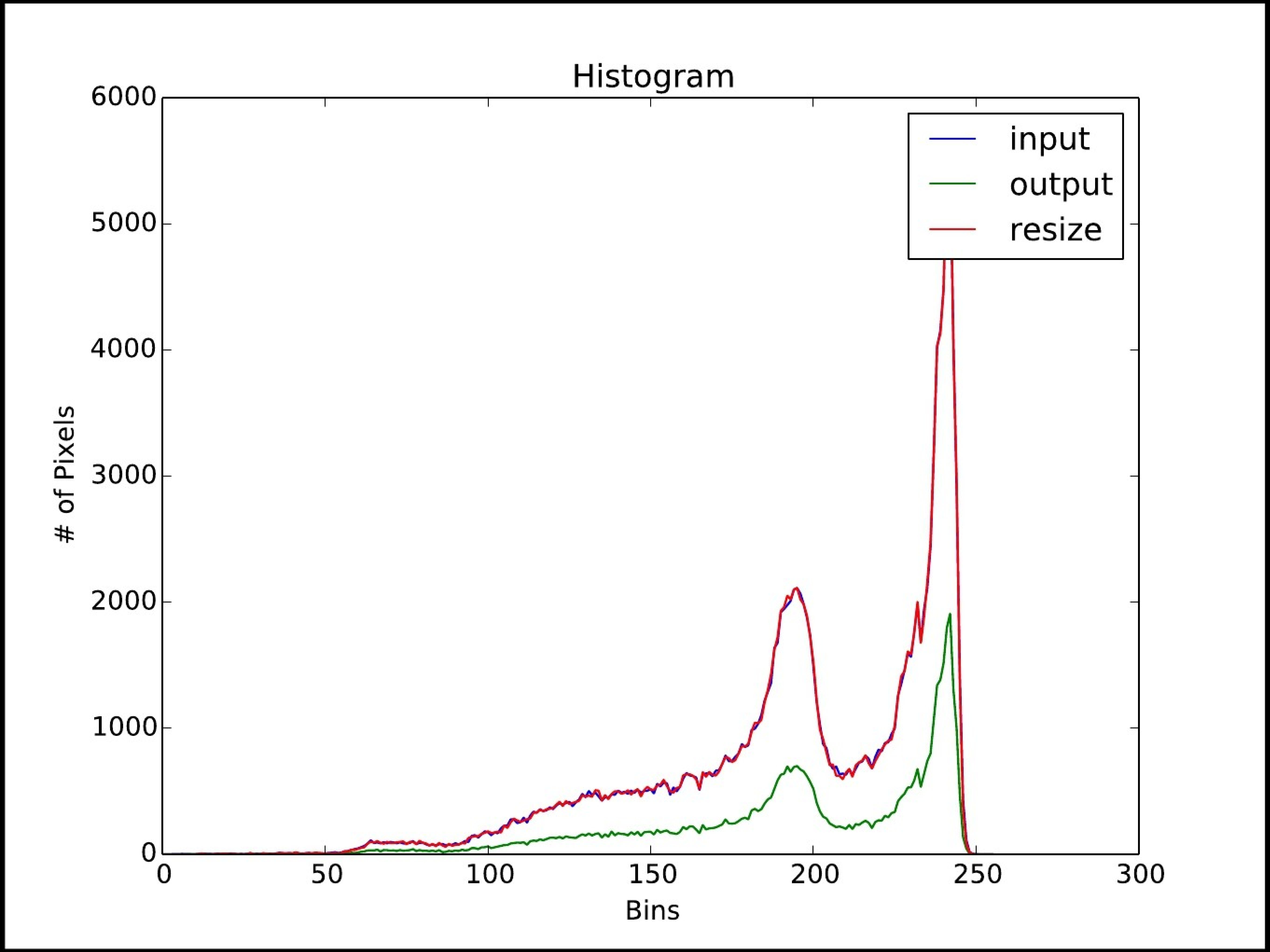
Input

Output



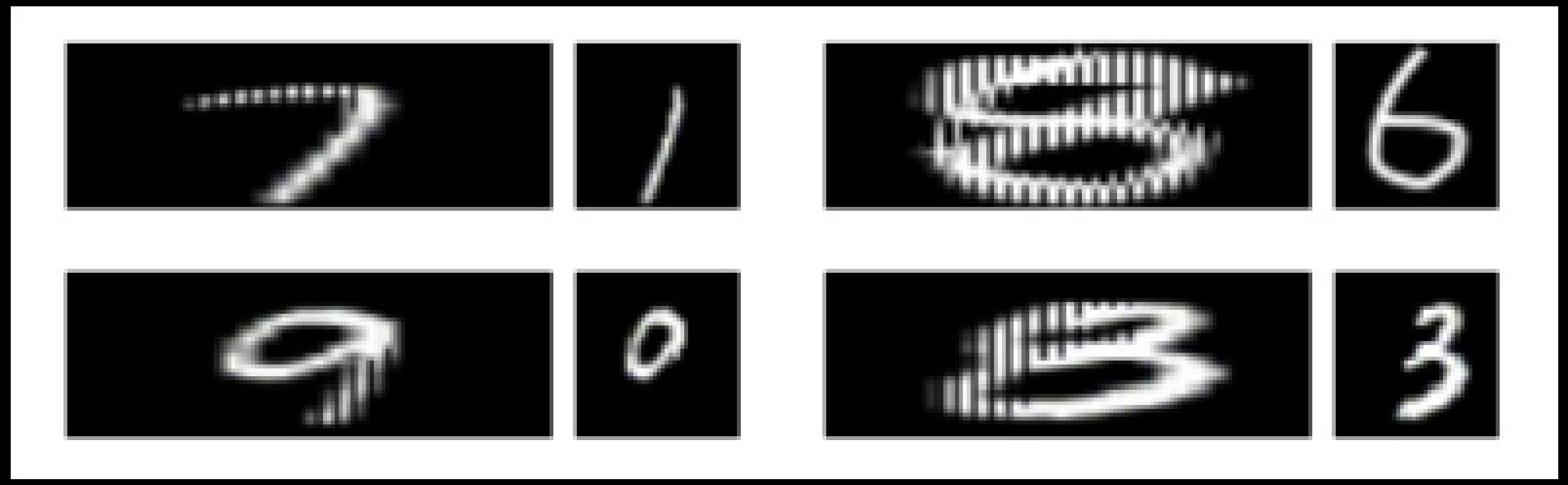
Resize





[illegible]

- [illegible]
- [illegible]
- [illegible]
- [illegible]



1ST DEEP LEARNING AND SECURITY WORKSHOP

May 24, 2018

co-located with the
39th IEEE Symposium on Security and Privacy

kangli.ctf@gmail.com